Imperial College London
Department of Computing

Machine Learning for Outlier Detection in Medical Imaging

Jeremy Hui-Min Tan

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of Imperial College London
April 2022

Abstract

Outlier detection is an important problem with diverse practical applications. In medical imaging, there are many diagnostic tasks that can be framed as outlier detection. Since pathologies can manifest in so many different ways, the goal is typically to learn from normal, healthy data and identify any deviations. Unfortunately, many outliers in the medical domain can be subtle and specific, making them difficult to detect without labelled examples. This thesis analyzes some of the nuances of medical data and the value of labels in this context. It goes on to propose several strategies for unsupervised learning. More specifically, these methods are designed to learn discriminative features from data of a single class. One approach uses divergent search to continually find different ways to partition the data and thereby accumulates a repertoire of features. The other proposed methods are based on a self-supervised task that distorts normal data to form a contrasting class. A network can then be trained to localize the irregularities and estimate the degree of foreign interference. This basic technique is further enhanced using advanced image editing to create more natural irregularities. Lastly, the same self-supervised task is repurposed for few-shot learning to create a framework for adaptive outlier detection. These proposed methods are able to outperform conventional strategies across a range of datasets including brain MRI, abdominal CT, chest X-ray, and fetal ultrasound data. In particular, these methods excel at detecting more subtle irregularities. This complements existing methods and aims to maximize benefit to clinicians by detecting fine-grained anomalies that can otherwise require intense scrutiny. Note that all approaches to outlier detection must accept some assumptions; these will affect which types of outliers can be detected. As such, these methods aim for broad generalization within the most medically relevant categories. Ultimately, the hope is to support clinicians and to focus their attention and efforts on the data that warrants further analysis.

Acknowledgements

So much has happened since I started this program and I owe a great deal to the people who have helped me at every step. My supervisor, Bernhard Kainz, has been a counsellor, guide, mentor, advocate and so much more than I deserve. He has gone above and beyond on many occasions and has never asked for anything in return other than (jokingly) requesting that we do more good work. I am not sure if I have held up my end of the bargain, but Bernhard has always celebrated every minor success and has always encouraged me to keep moving forward. I have heard that perseverance is a prerequisite for doing a PhD, but here I may have an unfair advantage with the endless patience and unfailing support of my supervisor. I will strive to express my gratitude by trying to do 'more good work'.

I am also very grateful to many faculty members and collaborators including Daniel Rueckert, Ben Glocker, Wenjia Bai, Antoine Cully, Thomas Day, and Ken'ichi Morooka. All of them have been very generous with their time and expertise. I also want to thank the Imperial staff and computing support group, including Amani El-Kholy, Katherine Bellenie, Geoff Bruce, and Lloyd Kamara. The college has also been more than generous in providing funding through the president's scholarship.

The BioMedIA lab is a wonderful community and I am grateful to every single member. It has been a real pleasure working alongside peers that have so much knowledge and skill as well as the humility to share it with me. I would like to thank Benjamin Hou, Qingjie Meng, Amir Alansary, Steven McDonagh, Loic Le Folgoc, Daniel Grzech, Daniel Coelho de Castro, James Batten, Huaqi Qiu, Turkay Kart, Tianrui Liu, and many more. I am honored to have been a part of this growing family. I also owe a special thanks to Martin Nowack for picking me up at my lowest point.

My family has always been a safety net that I can rely on, regardless of the state of the world or the state of my own affairs. I know that not everyone can say this and I am extremely privileged to have this bedrock, a home base that I can always return to. I am so grateful to my family and everyone else that has helped me, thank you.

‘There is no ceiling and it’s not a ladder.’

–*Comment on Vim by James Batten*

Contents

Chapter 1

Introduction

1.1 Motivation and Objectives

Machine Learning in Medical Image Analysis

Radiologists rely on medical imaging for rich diagnostic information. Having more diagnostically valuable data at their disposal can help clinicians to make more informed decisions. However, searching for clinically relevant details can become more strenuous as the volume of data increases. This is especially true when radiologists need to perform a broader search because i) the underlying condition is unknown or ii) its appearance is highly variable. Unfortunately, human limitations make it hard for radiologists to match the demand for speed and quality of diagnoses, even at the best institutions. Further challenges, such as lack of expertise, can amplify these issues when considering healthcare at the population level or in developing nations.

Machine learning offers a promising avenue for uniformly distributing clinical decision support. However, conventional methods generally require large amounts of labeled data for *each* desired task, which is expensive and time consuming to acquire. Another limitation is that these frameworks are primarily designed to recognize predefined disease classes. This can certainly be useful in reducing routine workload, but it cannot be used to detect unexpected ailments

or pathologies that have manifested in usual ways. Unfortunately, these incidents can be easy to miss because of biases such as inattentional blindness [Drew et al., 2013]. Having a clinical decision support tool that offers a complementary interpretation of the data could help in this regard. One potential strategy is to develop a tool that highlights all regions that deviate from normal, in any way. Rather than detecting predefined classes, this type of tool could act as an assistant, prompting clinicians to consider regions that warrant closer inspection. This way of framing the problem is sometimes called outlier detection.

Outlier Detection

Outlier detection can be used for many purposes ranging from automated quality assurance to disease screening. It is particularly useful when the characteristics of abnormal cases are not known *a priori*. This could be due to a lack of data from abnormal classes or difficulty in finding a common denominator between all types of defects. For example, rare diseases are defined as conditions that affect fewer than one in 2000 people; but it is estimated that rare diseases *collectively* affect one in every 17 people at some point in their lifetime [National Congenital Anomaly and Rare Disease Registration Service, 2021]. In these cases, it can be difficult to collect sufficient data on every disease and even human expertise may be scarce. To tackle these types of scenarios, outlier detection methods often assume that no training examples of outliers are available [Pimentel et al., 2014]. This is problematic for standard machine learning methods which rely on labelled examples.

The absence of abnormal examples creates several issues. Firstly, the network cannot learn characteristic features of outliers without being given examples during training. Another issue is the openness of the task. In a conventional supervised scenario, a network is tuned to identify features that are present in one class but not in the other. These contrasting features can also change depending on the opposing class. For example, features that are relevant for classifying pathologies may not be useful in recognizing imaging artifacts. Supervised methods have enjoyed widespread success by pinpointing these specific features and ignoring unimportant aspects. Unfortunately, it is difficult to achieve this without having a target class to contrast

with. In the case of outlier detection, any deviation could be important.

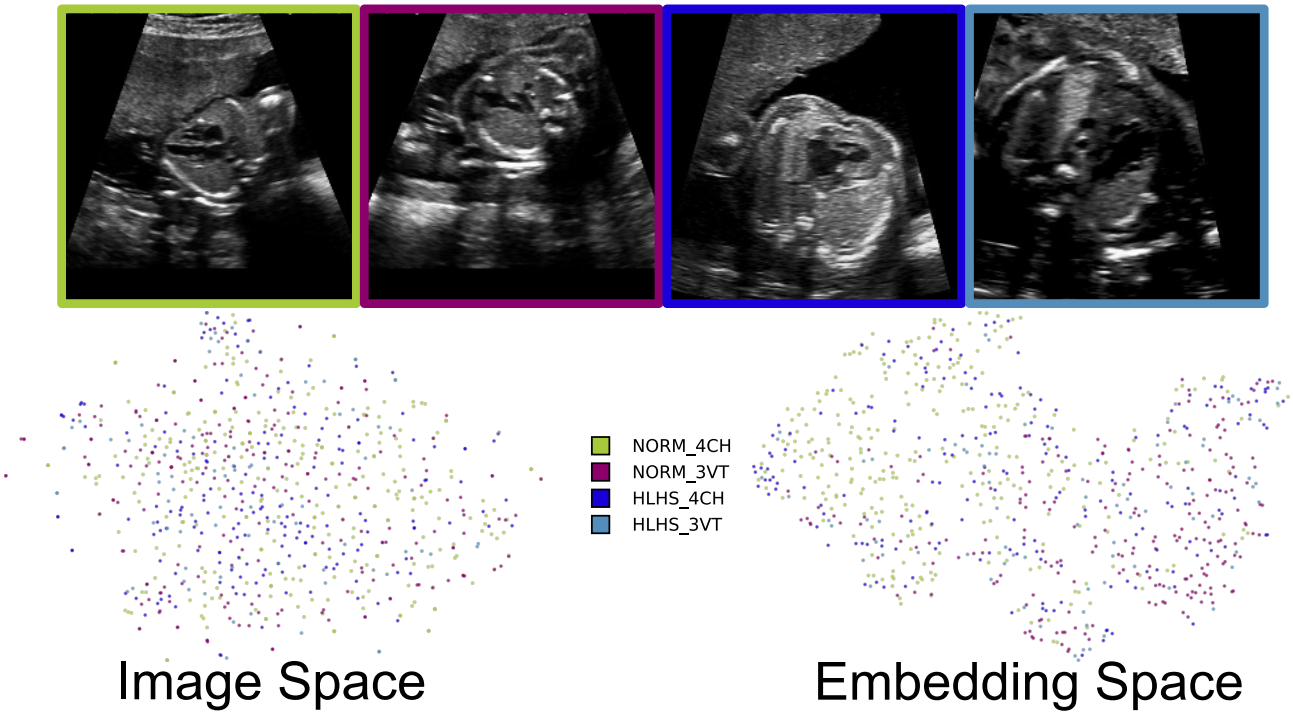


Figure 1.1: *t-SNE plots illustrating separability of different classes. Examples of each class are provided at the top following the legend color code. The t-SNE plot on the left uses samples in native input space. The right plot uses embeddings of a network trained to separate normal 4CH and 3VT views. Plots were produced using openTSNE [Poličar et al., 2019].*

Consider the example presented in Figure 1.1. This illustrates a real world application of outlier detection and highlights some of the key technical challenges involved.

The images at the top of Figure 1.1 are fetal ultrasound scans, specifically two important views of the heart, the four chamber view (4CH) and the three vessel and trachea view (3VT). Normal data is generally abundant, but conditions such as hypoplastic left heart syndrome (HLHS) can be very rare. Across all births in the U.K., only about 0.6% are affected by some form of congenital heart disease [National Congenital Anomaly and Rare Disease Registration Service, 2019]. With such low prevalence, it is difficult to expect sonographers across the country to be knowledgeable about every subtype. Nevertheless, detecting these conditions is important because congenital heart disease is associated with more perinatal and infant deaths than any other congenital abnormality [National Congenital Anomaly and Rare Disease Registration Service, 2019].

These factors make congenital heart disease a prime candidate for outlier detection. But there

are several aspects that make this type of problem very challenging. Consider the t-distributed stochastic neighbor embedding (t-SNE) plot using samples in image space (Figure 1.1). Because of the curse of dimensionality [Bellman, 2015] the distances between similar images and dissimilar images can be approximately the same. This makes it difficult to distinguish pathological samples from healthy ones in image space. To get around this, neural networks are often employed to map images to an embedding space. Ideally, this representation should emphasize semantic differences and disregard irrelevant information. This mapping is generally learned with labelled examples. For instance, the embedding space t-SNE plot in Figure 1.1 uses a network trained to separate normal 4CH and normal 3VT images. Even though this representation may involve important structures of the heart, it lacks the specificity required to cleanly separate healthy and pathological samples. The combination of these two problems forms the main dilemma in outlier detection: *input space has too much noise for outliers to stand out, and embeddings do not emphasize the right features unless they are trained with relevant samples, i.e. real outliers.* This conundrum is the motivation behind the works presented in the following chapters. Exploring this problem can lead to new solutions for specific applications in medical imaging and could also open up avenues for new approaches to generic machine learning.

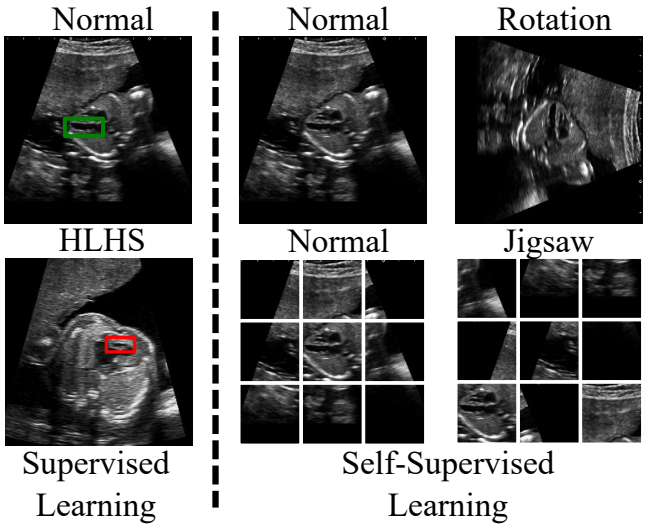


Figure 1.2: *Comparison between supervised and self-supervised learning. Supervised methods learn from labelled data, while self-supervised methods use auxiliary tasks to learn from unlabelled or normal data.*

As mentioned earlier, supervised methods are the most direct way to learn relevant features. These methods learn from labelled data, which can sometimes include detailed annotations.

For example, Figure 1.2 portrays samples with bounding boxes surrounding the left ventricle, a critical structure for the diagnosis of HLHS. Since labelled anomalies are not available in outlier detection settings, self-supervised learning is an attractive alternative. These methods effectively create their own labels by synthesizing auxiliary tasks. Figure 1.2 gives examples of two self-supervised tasks. The first approach applies different transformations, e.g. image rotation, to create new artificial classes [Golan and El-Yaniv, 2018]. Meanwhile the second example, jigsaw [Noroozi and Favaro, 2016], divides images into patches and permutes their positions creating artificial classes for each predetermined permutation. These methods were developed in the context of natural images and are mostly concerned with the overall structure of the image. This may be less helpful for diagnostic tasks, where global structure is generally preserved in healthy and pathological classes. Nevertheless, self-supervised methods can be designed in numerous ways and a major goal of this thesis is to devise objective functions that cater specifically to medical outlier detection.

1.2 Contributions

The work presented in this thesis explores a range of approaches for outlier detection. Techniques in semi-supervised learning, unsupervised learning, and self-supervised learning are considered. Although recent works have made significant progress in outlier detection for natural images, the nuances of medical data can sometimes stymie these methods. This thesis highlights some of the key challenges that are specific to medical outlier detection and proposes methods that are tailored to these issues.

The main contributions of this thesis are listed below. The key publications associated with each contribution are listed under each topic.

1. An exploration of the value of labels in medical imaging data

    - Tan, J., Au, A., Meng, Q., and Kainz, B. (2019). Semi-supervised learning of fetal anatomy from ultrasound. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pages 157–164. Springer

- Tan, J., Au, A., Meng, Q., FinesilverSmith, S., Simpson, J., Rueckert, D., Razavi, R., Day, T., Lloyd, D., and Kainz, B. (2020). Automated detection of congenital heart disease in fetal ultrasound screening. In *Medical Ultrasound, and Preterm, Perinatal and Paediatric Image Analysis*, pages 243–252. Springer

2. A divergent search algorithm for unsupervised representation learning

   - Tan, J. and Kainz, B. (2020). Divergent search for image classification behaviors. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 91–92. https://doi.org/10.1145/3377929.3389973

3. A self-supervised approach for outlier detection

   - Tan, J., Hou, B., Batten, J., Qiu, H., Kainz, B., et al. (2022). Detecting outliers with foreign patch interpolation. *Machine Learning for Biomedical Imaging*, 1(April 2022 issue):1–10

4. Enhanced methods for self-supervised outlier detection using image manipulation techniques and meta-learning

   - Tan, J., Hou, B., Day, T., Simpson, J., Rueckert, D., and Kainz, B. (2021a). Detecting outliers with poisson image interpolation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 581–591. Springer
   - Tan, J., Kart, T., Hou, B., Batten, J., and Kainz, B. (2021b). Metadetector: Detecting outliers by learning to learn from self-supervision. In *Biomedical Image Registration, Domain Generalisation and Out-of-Distribution Analysis at MICCAI*, pages 119–126. Springer

The first topic studies the value of labels in medical imaging applications, specifically fetal ultrasound. Data labelling in fetal ultrasound can be expensive because experts are in short supply [Van Velzen et al., 2016]. The first section investigates whether semi-supervised learning techniques, popularized in the context of natural images, can be applied successfully in fetal ultrasound. Factors such as class imbalance and high similarity between certain classes can pose challenges for these semi-supervised methods. Even when data is fully labelled, tasks such as classifying congenital heart disease can be difficult. The second section uses a multi-task learning approach to make the most of the available labels.

The next section proposes a divergent search strategy for learning features from unlabelled data. Unlike clustering methods that sort data into natural partitions, this approach continually searches for different ways to partition the data. This offers an alternative approach to discriminative feature learning for scenarios where the unlabelled data may only contain one class.

The third contribution is a self-supervised method for outlier detection. In the absence of contrasting classes or real outlier samples, this method synthesizes deviations to promote learning of normal features. To be specific, the network is trained to detect where and to what degree the normal features have been altered. During test time, this provides localization and an estimate of the degree of abnormality

The last section enhances the self-supervised approach from the previous section. The first part uses Poisson image editing [Pérez et al., 2003] to improve synthetic irregularities and reduce overfitting to artifacts of the self-supervised task. Part two uses the self-supervised task for few-shot learning in order to create an adaptive outlier detection method.

1.3 Statement of Originality

I declare that the work presented within this thesis is my own, unless appropriately referenced. Content that has been previously published is noted in the sections below.

1.4 Copyright Declaration

1.5 Publications

For completion, below is a list of the key publications associated with chapters in this thesis as well as additional publications.

1.5.1 Key Publications

Parts of this thesis have been published in the following articles.

[1] Tan, J., Au, A., Meng, Q., and Kainz, B. (2019). Semi-supervised learning of fetal anatomy from ultrasound. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pages 157–164. Springer

[2] Tan, J., Au, A., Meng, Q., FinesilverSmith, S., Simpson, J., Rueckert, D., Razavi, R., Day, T., Lloyd, D., and Kainz, B. (2020). Automated detection of congenital heart disease in fetal ultrasound screening. In *Medical Ultrasound, and Preterm, Perinatal and Paediatric Image Analysis*, pages 243–252. Springer

[3] Tan, J. and Kainz, B. (2020). Divergent search for image classification behaviors. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 91–92. https://doi.org/10.1145/3377929.3389973

[4] Tan, J., Hou, B., Batten, J., Qiu, H., Kainz, B., et al. (2022). Detecting outliers with foreign patch interpolation. *Machine Learning for Biomedical Imaging*, 1(April 2022 issue):1–10

[5] Tan, J., Hou, B., Day, T., Simpson, J., Rueckert, D., and Kainz, B. (2021a). Detecting outliers with poisson image interpolation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 581–591. Springer

[6] Tan, J., Kart, T., Hou, B., Batten, J., and Kainz, B. (2021b). Metadetector: Detecting outliers by learning to learn from self-supervision. In *Biomedical Image Registration, Domain Generalisation and Out-of-Distribution Analysis at MICCAI*, pages 119–126. Springer

1.5.2 Additional Publications

There are also several related publications which deal with other approaches to outlier detection, prerequisite tasks in fetal ultrasound, or other ways in which machine learning can be applied to medical imaging. They are listed below in reverse chronological order.

[1] Budd, S., Sinclair, M., Day, T., Vlontzos, A., Tan, J., Liu, T., Matthew, J., Skelton, E., Simpson, J., Razavi, R., et al. (2021). Detecting hypo-plastic left heart syndrome in fetal ultrasound via disease-specific atlas maps. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 207–217. Springer

[2] Chotzoglou, E., Day, T., Tan, J., Matthew, J., Lloyd, D., Razavi, R., Simpson, J., Kainz, B., et al. (2021). Learning normal appearance for fetal anomaly screening: Application to the unsupervised detection of hypoplastic left heart syndrome. *Machine Learning for Biomedical Imaging*, 1(September 2021 issue):1–10

[3] Schlüter, H. M., Tan, J., Hou, B., and Kainz, B. (2021). Self-supervised out-of-distribution detection and localization with natural synthetic anomalies (nsa). *arXiv preprint arXiv:2109.15222*

[4] Liu, T., Meng, Q., Vlontzos, A., Tan, J., Rueckert, D., and Kainz, B. (2020). Ultrasound video summarization using deep reinforcement learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 483–492. Springer

[5] Castro, D. C., Tan, J., Kainz, B., Konukoglu, E., and Glocker, B. (2019). Morpho-mnist: quantitative assessment and diagnostics for representation learning. *Journal of Machine Learning Research*, 20(178):1–29

[6] Folgoc, L. L., Castro, D. C., Tan, J., Khanal, B., Kamnitsas, K., Walker, I., Alansary, A., and Glocker, B. (2019). Controlling meshes via curvature: Spin transformations for pose-invariant shape processing. In *International Conference on Information Processing in Medical Imaging*, pages 221–234. Springer

[7] Tuysuzoglu, A., Tan, J., Eissa, K., Kiraly, A. P., Diallo, M., and Kamen, A. (2018). Deep adversarial context-aware landmark detection for ultrasound imaging. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 151–158. Springer

[8] Meng, Q., Baumgartner, C., Sinclair, M., Housden, J., Rajchl, M., Gomez, A., Hou, B., Toussaint, N., Zimmer, V., Tan, J., et al. (2018). Automatic shadow detection in 2d ultrasound images. In *Data Driven Treatment Response Assessment and Preterm, Perinatal, and Paediatric Image Analysis*, pages 66–75. Springer

Chapter 2

Background

In outlier detection, there are often constraints on the availability of data, particularly for anomalous classes. This poses several problems for data-driven methods. Nevertheless, overcoming the challenges of incomplete data is important for practical applications as well as theoretical advancements in machine learning. As such, one of the major goals throughout this work is to find new and useful training objectives. Different objective functions allow a model to learn from data in different ways. This can lead to more specialized models, tailored for certain tasks or models with better generalization across multiple tasks. Before exploring new objective functions, the major fundamental strategies are reviewed in the sections below. These include supervised, semi-supervised, unsupervised, and self-supervised learning strategies.

2.1 Supervised Learning: Foundations of Machine Learning

Supervised learning is a cornerstone of machine learning. Much of the success in modern deep learning has involved supervised learning in some capacity [LeCun et al., 2015]. While it is one of the most effective methods, it also demands the most data. Detailed labels help provide a clear objective and allow gradients to be computed for differentiable models. These gradients indicate

how the model weights can be adjusted to incrementally improve performance in the training task. One of the most common tasks in supervised learning is image classification. Through this lens, several key elements of a typical supervised learning pipeline will be discussed below.

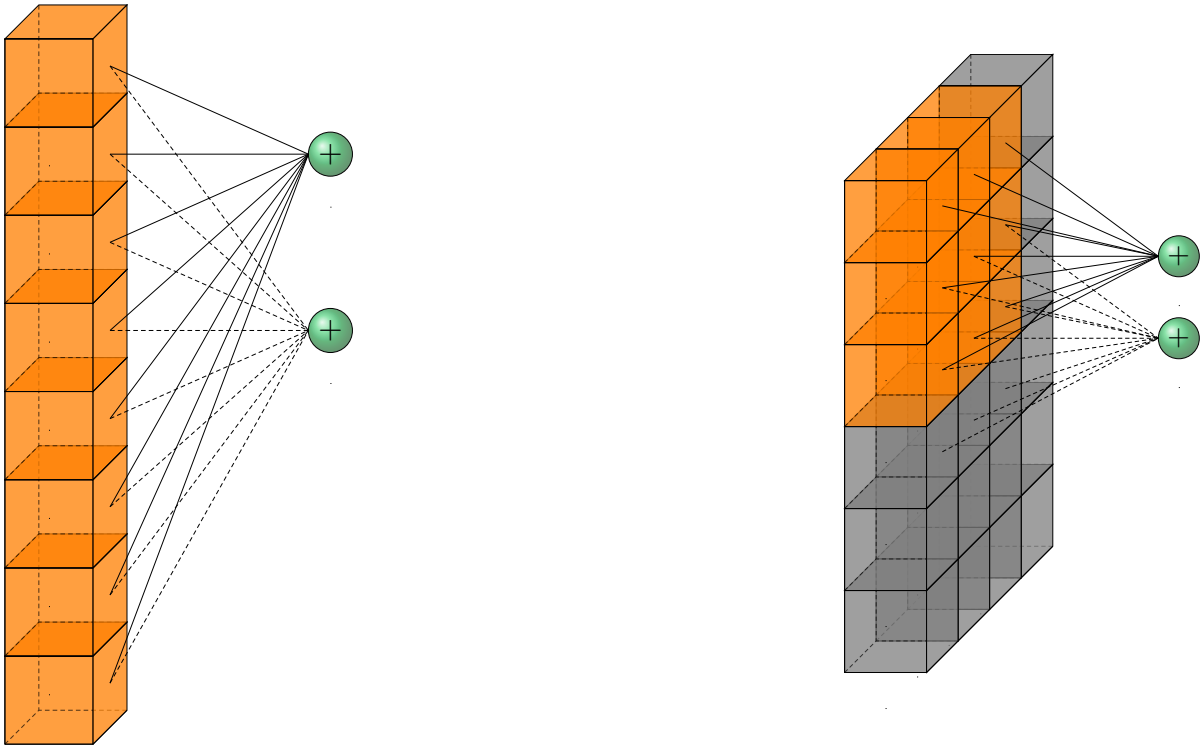
The Neuron: Before looking at a complete network, it can be helpful to briefly review the basic neuron unit. As their namesake suggests, these "artificial" neurons take inspiration from biological neurons which are cells that specialize in signal and information processing [Yuste, 2015]. Just as the brain and nervous system are composed of neurons, artificial neural networks are built by connecting neuron units in different configurations. In most cases, each neuron in the network follows the same basic operation, multiplying a set of inputs, $\boldsymbol{x} \in \mathbb{R}^d$, by a set of weights, $\boldsymbol{w} \in \mathbb{R}^d$, and summing them together (sometimes with a bias parameter, $b \in \mathbb{R}$). If included, b is typically a single scalar value per neuron. This basic operation is depicted in Eqn. 2.1 and Figure 2.1. In this equation $\boldsymbol{w}$ and $\boldsymbol{x}$ are vectors of equal dimension and $\boldsymbol{w} \cdot \boldsymbol{x}$ is a dot product, which could alternatively be written as a weighted sum of $\boldsymbol{x}$.

$$f(\boldsymbol{x}; \boldsymbol{w}, b) = \boldsymbol{w} \cdot \boldsymbol{x} + b \tag{2.1}$$

$$\begin{aligned} f_{net}(\boldsymbol{x}; \boldsymbol{W}, \boldsymbol{b}) &= f_3\Big(f_2\big(f_1(\boldsymbol{x}; \boldsymbol{W}_1, \boldsymbol{b}_1); \boldsymbol{W}_2, \boldsymbol{b}_2\big); \boldsymbol{W}_3, \boldsymbol{b}_3\Big) \\ &= \boldsymbol{W}_3\Big(\boldsymbol{W}_2\big(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1\big) + \boldsymbol{b}_2\Big) + \boldsymbol{b}_3 \end{aligned} \tag{2.2}$$

Each connection (line) in Figure 2.1 is associated with a weight parameter, w_i, and each neuron (green) is associated with a bias parameter, b_i. In a layer of fully connected neurons, all inputs to the layer are connected to all neurons within that layer (Figure 2.1.a). To represent a layer of h neurons, Eqn. 2.1 can be rewritten using $\boldsymbol{W} \in \mathbb{R}^{h \times d}$ and $\boldsymbol{b} \in \mathbb{R}^h$. The dot product then becomes a matrix multiplication, where each row of $\boldsymbol{W}$ contains the weights for one neuron.

Chaining multiple fully connected layers together in a sequence forms a network, as expressed in Eqn. 2.2. This arrangement can quickly lead to an enormous number of tunable parameters. This makes optimization computationally expensive and prone to overfitting. To remedy these issues, convolutional layers only connect each neuron to a small window of inputs (orange elements in Figure 2.1.b) and reuse connection weights (lines) by sliding the window across the entire input, i.e. convolving the input with a kernel of weights [Fukushima, 1980, Fukushima



(a) *Fully Connected Neuron* (b) *Convolutional Neuron*

Figure 2.1: *Example neurons in a fully connected MLP vs. kernels in a CNN. Graphic produced with PlotNeuralNet [Iqbal, 2018].*

and Miyake, 1982, LeCun et al., 1998]. The most basic 1D convolution operation is shown in Eqn. 2.3, where $\boldsymbol{w} \in \mathbb{R}^k$ is a kernel of size k and $b \in \mathbb{R}$ is a bias (one scalar per kernel) which is added to every element of the convolution output. For images, a 2D kernel is often used, e.g. the 3×3 kernel shown in Figure 2.1.b. In this illustration, the connection weights of the bottom neuron (dotted lines) are identical to the connection weights of the top neuron (solid lines) because the same kernel is being applied at each point in the image. This helps to exploit repetition of local patterns. Additional kernels provide capacity for more patterns and the outputs from each kernel form independent feature channels.

$$f_{\text{conv}}(\boldsymbol{x}; \boldsymbol{w}, b) = \boldsymbol{w} * \boldsymbol{x} + b \tag{2.3}$$

Both fully connected and convolutional layers are widely used to construct neural networks, sometimes referred to as multi-layer perceptrons (MLP's) [Werbos, 1994, McClelland et al., 1986] and convolutional neural networks (CNN's) [Fukushima, 1980, Fukushima and Miyake,

1982, LeCun et al., 1998], respectively. Modern configurations frequently combine different types of layers within the same network, so the nomenclature can be inexact and typically follows the majority.

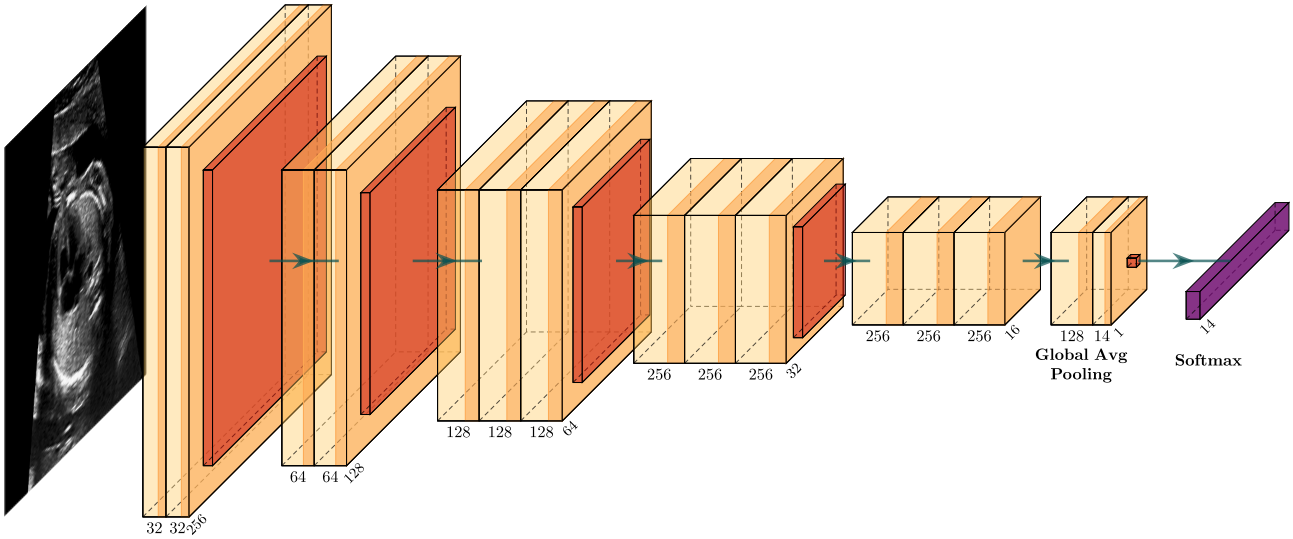


Figure 2.2: *Example neural network architecture, Sononet [Baumgartner et al., 2017]. The input is a fetal ultrasound image and the output is a probability distribution across 14 classes for different standard planes of fetal anatomy. The values below each block indicate the number of feature channels in each convolutional layer. Width and height dimensions are given by values at the corner of each block. Graphic produced with PlotNeuralNet [Iqbal, 2018].*

Architecture: Figure 2.2 gives an example of a contemporary CNN being used for medical ultrasound [Baumgartner et al., 2017]. CNN's are ubiquitous in many areas of machine learning, particularly those dealing with images. In addition to the fully connected and convolutional layers, briefly described above, most networks contain additional operations such as pooling and non-linear activations.

The pooling operation is designed to summarize information across the spatial dimensions, usually within small, non-overlapping windows. Successive pooling operations give CNN "encoders" their characteristic, narrowing shape (Figure 2.2). In practice, pooling can be performed by either i) increasing the stride in the convolution operations or ii) using dedicated pooling layers. In a pooling layer, patches from the input are replaced by the mean or maximum of the patch across spatial dimensions, effectively downsampling the entire input. As the spatial dimensions shrink with each pooling layer, the number of feature channels in the convolutional layers generally increases. This overall structure supports the construction of hierarchical fea-

tures [Krizhevsky et al., 2012]. Small kernels (e.g. 3x3 pixels) in shallow layers can identify low level features, such as lines or corners. Meanwhile deeper layers aggregate these fragments to identify larger shapes and composite patterns.

Non-linearities are another essential component of neural networks. Without them, even a deep neural network only amounts to a linear combination of the inputs. By introducing non-linearities, each layer adds more expressivity, allowing the network to learn more complex mappings between input and output. In fact, non-linear activations can even turn simple neural networks into universal function approximators, meaning they can approximate any continuous function arbitrarily well, provided certain conditions are met. This is explained more precisely as an aside in the box below.

Universal Approximation Theorem

Consider a network with a single hidden fully connected layer (input-hidden-output). Let $\sigma \in C(\mathbb{R})$ denote a continuous activation function used for all hidden units in the network. Observing these constraints means considering the set given in Eqn. 2.4. Note the linear span, which includes all linear combinations, represents an output neuron with weights for each neuron in the hidden layer.

$$\mathcal{M}(\sigma, \boldsymbol{w}, b) = \text{span}\{\sigma(\boldsymbol{w} \cdot \boldsymbol{x} + b) : \boldsymbol{w} \in \mathbb{R}^n, b \in \mathbb{R}\} \tag{2.4}$$

Provided there are enough neurons in the hidden layer, this network can approximate any continuous function arbitrarily well, on an arbitrary compact (closed and bounded) subset $X \subseteq \mathbb{R}^n$, i.e. $\mathcal{M}(\sigma, \boldsymbol{w}, b)$ is dense in the space $C(\mathbb{R}^n)$, if and only if σ is not a polynomial [Leshno et al., 1993]. This version of the theorem builds on seminal work by several others and this remains an area of active research. Further details can be found within their papers [Hornik, 1991, Cybenko, 1989, Funahashi, 1989, Hornik et al., 1989, Pinkus, 1999, Chong, 2020].

Practically speaking, common activation functions include sigmoid (Eqn. 2.5), hyperbolic tangent (Eqn. 2.6), rectified linear units (ReLU's) [Fukushima, 1980, Nair and Hinton, 2010] (Eqn. 2.7), and variations of these functions. ReLU's are among the most widely used ac-

tivation functions for hidden layers because the derivative of the ReLU function has properties that are beneficial for deeper networks. This will be explained in more detail in the section on backpropagation and weight initialization.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2.5}$$

$$\tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1} \tag{2.6}$$

$$\phi(z) = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.7}$$

Backpropagation and Parameter Initialization: The overall architecture described above will not yield a sensible mapping until its parameters are tuned for the desired task. This section covers the basics of how these parameters are updated and initialized.

Backpropagation is a technique for efficiently optimizing differentiable neural networks [Linnainmaa, 1970, Werbos, 1982, Rumelhart et al., 1985]. It works by computing gradients, which indicate how the objective function changes with respect to each parameter, and then updating these parameters by following the gradient in a direction that minimizes the objective function. Recall that Eqn. 2.1 defines the basic neuron operation as a function; in the same way, an entire network is a composite function, mapping inputs to outputs through a series of neuron operations and activations. The rate of change of the objective function with respect to a parameter in the network can thus be found by differentiating with the chain rule.

Backpropagation

Consider a fully connected network with L layers, q output neurons, and an activation function σ used throughout the network. Let $z_j^{(l)}$ and $a_j^{(l)}$ denote the output of neuron j, in layer l, before and after the activation function, respectively.

$$a_j^{(l)} = \sigma(z_j^{(l)}) \tag{2.8}$$

$$\begin{aligned} z_j^{(l)} &= \boldsymbol{w}_j^{(l)} \cdot \boldsymbol{a}^{(l-1)} + b_j^{(l)} \\ &= w_{j1}^{(l)} a_1^{(l-1)} + w_{j2}^{(l)} a_2^{(l-1)} + ... + b_j^{(l)} \end{aligned} \tag{2.9}$$

The objective function, $\mathcal{J}$, can then be evaluated by comparing each output neuron, $a_j^{(L)}$, with the corresponding target value, y_j, (for a given sample) and averaging across all n samples in the dataset. In this example, mean squared error is used (Eqn. 2.10).

$$\mathcal{J} = \frac{1}{n}\sum_{m=1}^{n}\mathcal{J}_m = \frac{1}{n}\sum_{m=1}^{n}\frac{1}{q}\sum_{j=1}^{q}(a_{mj}^{(L)} - y_{mj})^2 \tag{2.10}$$

Evaluating the objective function gives a measure of the network's performance with the current parameters. Since this function is differentiable, the gradient provides an estimate of how each parameter should be changed to improve performance. Below are a few examples of partial derivatives with respect to different parameters. For simplicity, only a single sample is shown and the m subscripts are omitted.

$$\begin{aligned}\frac{\partial \mathcal{J}}{\partial w_{jk}^{(L)}} &= \frac{\partial z_j^{(L)}}{\partial w_{jk}^{(L)}}\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}}\frac{\partial \mathcal{J}}{\partial a_j^{(L)}} \\ &= \left(a_k^{(L-1)}\right)\left(\sigma'(z_j^{(L)})\right)\left(\frac{2}{q}(a_j^{(L)} - y_j)\right)\end{aligned} \tag{2.11}$$

$$\begin{aligned}\frac{\partial \mathcal{J}}{\partial b_j^{(L)}} &= \frac{\partial z_j^{(L)}}{\partial b_j^{(L)}}\frac{\partial a_j^{(L)}}{\partial z_j^{(L)}}\frac{\partial \mathcal{J}}{\partial a_j^{(L)}} \\ &= \left(1\right)\left(\sigma'(z_j^{(L)})\right)\left(\frac{2}{q}(a_j^{(L)} - y_j)\right)\end{aligned} \tag{2.12}$$

$$\begin{aligned}\frac{\partial \mathcal{J}}{\partial w_{jk}^{(L-1)}} &= \frac{\partial z_j^{(L-1)}}{\partial w_{jk}^{(L-1)}}\frac{\partial a_j^{(L-1)}}{\partial z_j^{(L-1)}}\frac{\partial \mathcal{J}}{\partial a_j^{(L-1)}} \\ &= \frac{\partial z_j^{(L-1)}}{\partial w_{jk}^{(L-1)}}\frac{\partial a_j^{(L-1)}}{\partial z_j^{(L-1)}}\sum_{i=1}^{q}\frac{\partial z_i^{(L)}}{\partial a_j^{(L-1)}}\frac{\partial a_i^{(L)}}{\partial z_i^{(L)}}\frac{\partial \mathcal{J}}{\partial a_i^{(L)}} \\ &= \left(a_k^{(L-2)}\right)\left(\sigma'(z_j^{(L-1)})\right)\left(\sum_{i=1}^{q} w_{ij}^{(L)}\cdot\sigma'(z_i^{(L)})\cdot\frac{2}{q}(a_i^{(L)} - y_i)\right)\end{aligned} \tag{2.13}$$

The notation for the weights follows from Eqn. 2.9, e.g. $w_{jk}^{(l)}$ corresponds to the weight value connecting neuron k in layer $l-1$ to neuron j in layer l. Eqn. 2.13 gives an example of backpropagation through multiple layers. Since the weight $w_{jk}^{(L-1)}$ in layer $L-1$ affects all output neurons in layer L, the error backpropagation for this weight must take all of these neurons into account. This is shown as a summation of the partial derivatives for q output neurons. As more layers are involved, the number of terms in

the equation grows rapidly. However, backpropagation exploits the reoccurring terms to keep the computation efficient, even with deeper networks.

The equations above can be extended to compute the gradient for all parameters in the network. Using this gradient, the network parameters can be updated via gradient descent [Cauchy et al., 1847], as shown in Eqn. 2.14. Note that ∇ represents the gradient and η is a learning rate that controls the step size of the optimization process. Also note that the gradient term is subtracted because the gradient points in the direction of steepest *ascent*. Gradient descent normally computes the gradient using the entire set of training samples, x, and labels, y. However many datasets are too large for this to be practical. Instead, stochastic gradient descent [Robbins and Monro, 1951] is widely used to approximate the gradient using mini-batches of samples.

$$\theta_{i+1} = \theta_i - \eta \nabla_\theta \mathcal{J}(\theta_i; x, y) \tag{2.14}$$

The parameter optimization process begins with initialization. Since gradients are dependent on the current parameter values, the initialization of the network becomes a matter of critical importance. The backpropagation equations also provide several insights for effective parameter initialization. For instance, consider neurons in a fully connected layer that all receive the same inputs and feed into the same outputs. If the weights in these neurons are initialized with identical values, then their updates and influence on other parameters will mirror each other throughout the optimization. To make better use of the parameters, the initialization should break these "symmetries" and allow each neuron to learn a different feature. Also, partial derivatives across multiple layers (e.g. Eqn. 2.13) involve the product of activation values and the weight values themselves. Therefore, if the magnitude of the weight values is too large or too small, then their product across several layers may lead to a gradient that is too large or small, respectively. Lastly, the derivative of the activation function, σ, is another important term in the partial derivative to consider. For example, the sigmoid function saturates for very small or very large numbers, which means that the gradient in these cases will be nearly zero. Conversely, if only the linear region of the sigmoid is used, then it limits the range of functions the network can learn and defeats the purpose of having non-linear activation functions.

Motivated by these considerations, several initialization schemes have been developed. Below is

a summary of the derivation for one approach, Xavier initialization [Glorot and Bengio, 2010].

Parameter Initialization

Consider a fully connected network with tanh activation functions used throughout. Let n_l denote the number of neurons in layer l. Assume that the weights are initialized independently and that all inputs have the same variance, $\text{Var}[\boldsymbol{x}]$. For simplicity, also assume that the weights are initialized within the linear region of the tanh activation function and that biases are initialized to zero. The variance of the first layer's post-activation output can be seen in Eqn. 2.15. Note that each weight term is simplified following Eqn. 2.16. Also, recall that the weights are initialized around the linear region of the tanh function, so any expected values, i.e. in Eqn. 2.16, will become zero. In the final line of Eqn. 2.15, the individual weight terms are summarized into a shared scalar variance term, $\text{Var}\big[w^{(1)}\big]$.

$$\begin{aligned}
\text{Var}\big[a^{(1)}\big] &\approx \text{Var}\big[z^{(1)}\big] \\
&= \text{Var}\big[\boldsymbol{w}^{(1)} \cdot \boldsymbol{x} + b^{(1)}\big] \\
&= \text{Var}\big[w_0^{(1)} x_0\big] + ... + \text{Var}\big[w_{n_1}^{(1)} x_{n_1}\big] + \text{Var}\big[b^{(1)}\big] \\
&= n_1 \text{Var}\big[w^{(1)}\big]\text{Var}\big[x\big]
\end{aligned} \tag{2.15}$$

$$\begin{aligned}
\text{Var}\big[w_j^{(1)} x_j\big] &= \mathbb{E}\big[w_j^{(1)}\big]^2 \text{Var}\big[x_j\big] + \mathbb{E}\big[x_j\big]^2 \text{Var}\big[w_j^{(1)}\big] + \text{Var}\big[w_j^{(1)}\big]\text{Var}\big[x_j\big] \\
&= \text{Var}\big[w_j^{(1)}\big]\text{Var}\big[x_j\big]
\end{aligned} \tag{2.16}$$

A similar expression can be derived in the opposite direction, going backwards from the output of the network (Eqn. 2.17).

$$\begin{aligned}
\text{Var}\left[\frac{\partial \mathcal{J}}{\partial z_j^{(L-1)}}\right] &= \text{Var}\left[\frac{\partial a_j^{(L-1)}}{\partial z_j^{(L-1)}} \sum_{i=1}^{n_L} \frac{\partial z_i^{(L)}}{\partial a_j^{(L-1)}} \frac{\partial a_i^{(L)}}{\partial z_i^{(L)}} \frac{\partial \mathcal{J}}{\partial a_i^{(L)}}\right] \\
&= \text{Var}\left[\Big(\sigma'(z_j^{(L-1)})\Big)\left(\sum_{i=1}^{n_L} w_{ij}^{(L)} \cdot \sigma'(z_i^{(L)}) \cdot \frac{\partial \mathcal{J}}{\partial a_i^{(L)}}\right)\right] \\
&= n_L \text{Var}\big[w^{(L)}\big]\text{Var}\left[\frac{\partial \mathcal{J}}{\partial a^{(L)}}\right]
\end{aligned} \tag{2.17}$$

Glorot and Bengio generalize the above expressions to compute the variance for any

layer in the network, as seen in Eqn. 2.18 and Eqn. 2.19 [Glorot and Bengio, 2010].

$$\mathrm{Var}\left[a^{(l)}\right] = \mathrm{Var}\left[x\right] \prod_{i=1}^{l} n_i \mathrm{Var}\left[w^{(i)}\right] \tag{2.18}$$

$$\mathrm{Var}\left[\frac{\partial \mathcal{J}}{\partial z^{(l)}}\right] = \mathrm{Var}\left[\frac{\partial \mathcal{J}}{\partial a^{(L)}}\right] \prod_{i=l}^{L} n_{i+1} \mathrm{Var}\left[w^{(i)}\right] \tag{2.19}$$

Based on these variance equations, Glorot and Bengio propose that an effective initialization scheme should satisfy the conditions in Eqn. 2.20 and Eqn. 2.21, or as compromise, Eqn. 2.22. This helps to keep a similar level of variance for the activations and gradients of each layer. This in turn aims to prevent vanishing or exploding gradients.

$$\forall\, l,\ n_l \mathrm{Var}\left[w^{(l)}\right] = 1 \tag{2.20}$$

$$\forall\, l,\ n_{l+1} \mathrm{Var}\left[w^{(l)}\right] = 1 \tag{2.21}$$

$$\forall\, l,\ \mathrm{Var}\left[w^{(l)}\right] = \frac{2}{n_l + n_{l+1}} \tag{2.22}$$

Assume a uniform distribution is used to sample weight values, $w \sim U[-\varepsilon, \varepsilon]$. Then the variance of w can be expressed in terms of ε (Eqn. 2.23). Note that the distribution is zero centered, so the expectation term evaluates to zero. Combining Eqn. 2.22 with Eqn. 2.23 gives the value of ε that fulfills the desired conditions.

$$\begin{aligned} \mathrm{Var}\left[w\right] &= \mathbb{E}\left[w^2\right] - \mathbb{E}\left[w\right]^2 \\ &= \frac{1}{2\varepsilon} \int_{-\varepsilon}^{\varepsilon} w^2 \, dw \\ &= \frac{1}{6\varepsilon} w^3 \Big]_{-\varepsilon}^{\varepsilon} \\ &= \frac{\varepsilon^2}{3} \end{aligned} \tag{2.23}$$

$$\varepsilon = \frac{\sqrt{6}}{\sqrt{n_l + n_{l+1}}} \tag{2.24}$$

The strategy described above is one instance of Xavier initialization [Glorot and Bengio, 2010]. Similar derivations can also be performed using a normal distribution. Furthermore, there are schemes that consider other types of activation functions, e.g. He initialization which caters to

ReLU non-linearities [He et al., 2015].

Objective Function: As described in the last section, the error defined by the objective function is backpropagated through the network to update the parameters. In this way, the objective function dictates how the network learns from the data. Designing the objective function is particularly important in unsupervised learning and outlier detection. Below is a brief description of a standard cross-entropy loss.

One of the most common configurations in contemporary machine learning is multi-class classification with a cross-entropy loss. These networks are typically terminated with a softmax activation (Eqn. 2.25). This operation converts raw logits, z_i, into class probabilities that sum to 1. The cross-entropy loss is given in Eqn. 2.26, where p is the true class probability distribution and q is the predicted class probability distribution. For a given input, p is the corresponding label (usually provided as a one-hot vector) and q is the distribution given by the softmax output of the neural network. If p is a one-hot vector with a value of 1 for class j and 0 otherwise, then Eqn. 2.26 reduces to $-log(q(j))$. This approaches 0 as $q(j)$ approaches 1. In this way, minimizing the cross-entropy loss aligns the predicted distribution to the ground truth distribution.

$$\sigma(z)_i = \frac{\mathrm{e}^{z_i}}{\sum_{j=1}^{K} \mathrm{e}^{z_j}} \tag{2.25}$$

$$H(p, q) = -\sum p(i) \log(q(i)) \tag{2.26}$$

More generally, cross-entropy can be thought of as the amount of surprise in the outcome of an event with probability distribution p, assuming that the outcome is governed by probability distribution q. Eqn. 2.27 expresses cross-entropy in terms of the inherent entropy of the true distribution, $H(p)$, and the relative entropy, or Kullback-Leibler (KL) divergence, of the two distributions.

$$H(p, q) = H(p) + D_{KL}(p\|q) \tag{2.27}$$

$$H(p) = -\sum p(i) \log(p(i)) \tag{2.28}$$

$$\begin{aligned} D_{KL}(p\|q) &= \sum p(i)\big(\log(p(i) - \log(q(i))\big) \\ &= \sum p(i) \log \frac{p(i)}{q(i)} \end{aligned} \tag{2.29}$$

Note that entropy is a measure of the uncertainty in a set of possible outcomes. Put another way, it is the average amount of information that is gained from drawing a sample from a probability distribution. If a probability distribution is skewed such that some outcomes are much more likely than others, then the information gained from drawing a sample is low (entropy close to 0). Conversely, entropy is maximized when the outcome is least predictable, e.g. a fair coin. In many cases, the entropy of the true distribution, $H(p)$, remains unchanged throughout the optimization. Furthermore, the network parameters being optimized typically have no influence on this term. On the other hand, the KL divergence term directly relates to the difference between the true and predicted distributions. Specifically, it is the expectation of the difference between the log probabilities of both distributions (Eqn. 2.29). Note that the KL divergence is not symmetric, i.e. $D_{KL}(p\|q) \neq D_{KL}(q\|p)$, and therefore cannot be considered a distance metric in the geometric sense. Also, the KL divergence is only defined whenever $q(i) \neq 0$ or if both $q(i) = 0$ and $p(i) = 0$ in which case $D_{KL}(p\|q) = 0$.

Cross-entropy, KL divergence, and mean squared error are among the most frequently used objective functions in contemporary machine learning. The following sections present different applications where these loss functions are used outside of supervised learning.

2.2 Semi-Supervised Learning: Exploiting Structure in the Data

The most straightforward variation on supervised learning is semi-supervised learning. It is used in scenarios where there is a limited number of labelled examples and a large amount of unlabelled data. The overall aim in this setting is to learn from the structure of the data and use it to infer labels for the unlabelled data. There are several strategies to accomplish this, many of which involve a standard supervised loss on labelled data and a consistency loss on the unlabelled data.

One of the highest performing semi-supervised methods, known as unsupervised data augmentation (UDA) [Xie et al., 2020], is depicted in Figure 2.3. Labelled samples, x_l, come from $\mathcal{D}_L$, the distribution of labelled data and are accompanied by ground truth labels, y^*. These are used with a standard cross-entropy loss to provide gradients for the desired classification task. Meanwhile, the consistency loss involves unlabelled samples, x_u, which are drawn from the distribution of unlabelled data, $\mathcal{D}_U$.

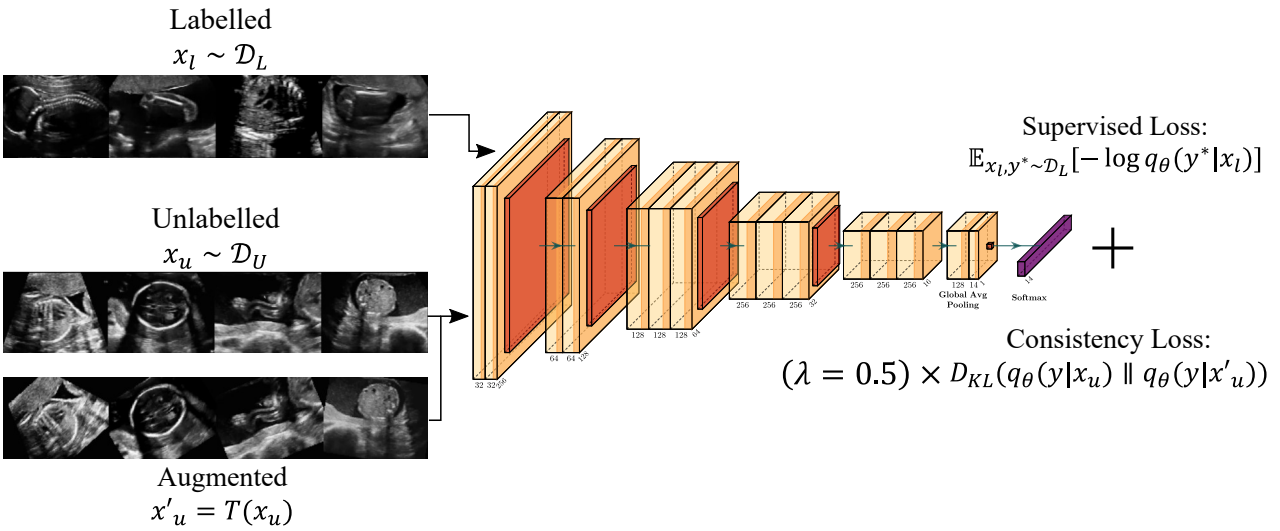


Figure 2.3: *UDA [Xie et al., 2020] consistency loss applied to Sononet [Baumgartner et al., 2017]. A standard supervised loss is used for labelled data. The consistency loss is given by the KL divergence between unlabelled samples and their augmentations. Graphic produced with PlotNeuralNet [Iqbal, 2018].*

In general, consistency losses are designed to minimize the KL divergence between two pre-

dictions on unlabelled data. The first prediction is usually performed with standard inference using the current model parameters and an unaltered sample. This is the path used for back-propagation. The second prediction is performed under different conditions, with a change in the network, the input, or both. This prediction is often used as a label and no gradients flow through this pathway. Some methods aim to make the second prediction less noisy in hopes of a more robust "label". For example, the mean teacher approach uses an exponential moving average of the network parameters to make the second prediction [Tarvainen and Valpola, 2017]. Other methods, such as temporal ensembling, average the predictions themselves over time [Laine and Aila, 2017]. The temporal ensembling method also proposes random augmentations to the inputs for the second prediction. This helps the network to learn a representation that is invariant to class preserving transformations. In a similar way, virtual adversarial training [Miyato et al., 2018b] computes gradients for the inputs to find noise patterns that elicit the greatest change in the network predictions. This additive noise, which the network is most sensitive to, is then used to improve the robustness of the predictions. UDA follows in the same vein, using targeted input augmentations. But rather than computing gradients for the inputs, UDA finds a specific set of augmentations, through AutoAugment [Cubuk et al., 2019] or RandAugment [Cubuk et al., 2020], that are deemed effective for the supervised task [Xie et al., 2020]. Figure 2.3 illustrates this approach where unlabelled samples are augmented through a set of transformations T. In this figure, θ is used to represent the network's current parameters and y is used to represent the network's prediction for a given sample, i.e. $y = f_\theta(x)$. As such, the UDA consistency loss, specified in Figure 2.3, uses the same parameters, but different inputs for the two predictions. Although the complete implementation is a bit more nuanced, the overall method is very simple and effective.

While many semi-supervised methods achieve excellent performance, it is important to analyze the conditions in which they succeed. Oliver et al. take note of the fact that nearly all of these methods use fully labelled datasets and simply withhold a certain portion of the labels [Oliver et al., 2018b]. This means that the "unlabelled" data is in fact curated data which falls into the predetermined classes with even distribution. This is very unlikely to occur in reality and in a sense betrays the definition of unlabelled data which is supposed to be uncurated and

unknown in nature. In their experiments, Oliver et al. create different sets of unlabelled data that are increasingly mismatched to the ideal (labelled) data distribution. They demonstrate that mismatched data not only fails to help, but can actually harm performance (compared to using no unlabelled data at all). This raises concerns that semi-supervised methods are fundamentally unable to take advantage of genuinely unlabelled data.

Ultimately, semi-supervised methods present many useful concepts; however careful attention must be paid to the conditions in which they are deployed. Indeed, the conditions of outlier detection are distinct from typical semi-supervised settings. Outlier detection problems generally only have one class available during training and the test data can come from any unknown distribution. Learning under these conditions is often done using unsupervised objective functions, which are reviewed in the following section.

2.3 Self-supervised and Unsupervised Representation Learning: Crafting Useful Objectives

Unsupervised learning is a broad term for nearly any type of learning that is done without using labels. It includes traditional methods such as clustering algorithms and autoencoders as well as more modern techniques including generative models and self-supervised tasks.

Recently, one class of self-supervised methods has renewed interest in unsupervised learning as a competitive alternative to supervised learning. These methods use a contrastive loss to compare representations of different images and push similar images closer together while spreading dissimilar images further apart. One of the most popular implementations is a simple framework for contrastive learning of visual representations, abbreviated as SimCLR [Chen et al., 2020b]. It involves a dataset of unlabelled images, $x \sim \mathcal{D}_U$, a set of random transformations transformations, $t \sim \mathcal{T}$, a convolutional (encoder) network, $h = f(x)$, and a smaller fully connected network, $z = g(h)$. Recall the consistency loss from the previous section on semi-supervised learning (Section 2.2). The consistency loss takes unlabelled samples, applies augmentations,

and minimizes the KL divergence between the network's predictions for these samples. SimCLR works in a very similar way, but with some key differences. In the semi-supervised setting, the network benefits from a supervised loss with known classes. This allows for the use of a softmax activation function which converts output logits into class probabilities, the required format for the KL divergence. In unsupervised learning, the classes are not known a priori, even the number of classes may be unknown. As such, instead of using outputs shaped for a specific number of classes, contrastive methods directly compare representation vectors which can have arbitrary dimensions. The metric used in SimCLR is cosine similarity (Eqn. 2.30), i.e. the dot product between two normalized vectors. This cosine similarity function is used to compare positive and negative pairs. A positive pair consists of two random and different augmentations of the same original image. Meanwhile a negative pair is any two augmented images that do not come from the same original image. To increase the similarity between positive pairs and decrease similarity between negative pairs, SimCLR uses each pair's similarity as an input to a softmax function. This is shown in Eqn. 2.31. Note that n original samples are augmented in two different ways to create $2n$ images in a batch. In this equation, the similarity between samples i and j appears in the numerator, while the denominator is the sum of the similarities of all other pairings (except for comparing sample i with itself, hence the indicator function $\mathbb{1}_{[k\neq i]}$). Note that τ is a temperature scaling parameter. The negative log in this equation is from the formula for cross-entropy. Recall from Eqn. 2.26 that cross-entropy is the sum of the product of i) the true probabilities and ii) the negative log of the predicted probabilities; in this case the latter is represented by $\ell(i,j)$. The former can be implicitly set to one for positive pairs and zero for all other pairs, resulting in Eqn. 2.32. Written this way, it is sometimes called the multinomial logistic loss. Note that for every k, the indices $2k-1$ and $2k$ correspond to two augmentations of the same sample, i.e. a positive pair.

$$\text{sim}(\boldsymbol{u},\boldsymbol{v}) = \frac{\boldsymbol{u}^\intercal \boldsymbol{v}}{\|\boldsymbol{u}\|\|\boldsymbol{v}\|} \tag{2.30}$$

$$\ell(i,j) = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i,\boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2n} \mathbb{1}_{[k\neq i]} \exp(\text{sim}(\boldsymbol{z}_i,\boldsymbol{z}_k)/\tau)} \tag{2.31}$$

$$\mathcal{L} = \frac{1}{2n}\sum_{k=1}^{n}\left[\ell(2k-1,2k) + \ell(2k,2k-1)\right] \tag{2.32}$$

As mentioned earlier, minimizing this contrastive loss will pull augmentations of the same image closer together and push other images further apart. After training with this loss, the smaller fully connected network can be discarded. A simple linear classifier can then be trained on top of the main encoder, using the representation vector, h, as an input and giving outputs customized to the desired task.

This method was inspired by contrastive predictive coding methods [Oord et al., 2018, Hénaff et al., 2019], which use patches from the same image as positive pairs and patches from different images for negative pairs. Contrastive losses have also been used much earlier in the context of similarity and metric learning [Hadsell et al., 2006].

There are also unsupervised methods that combine modern neural networks with more traditional clustering algorithms. DeepCluster [Caron et al., 2018] uses a similar setup to the self-supervised method described above with a convolutional encoder network, $h = f(x)$, and a smaller fully connected network, $z = g(h)$. DeepCluster takes a set of unlabelled images and uses the encoder network to obtain a representation, h, for each sample. A clustering algorithm known as k-means clustering [MacQueen et al., 1967, Lloyd, 1982] is then performed on the set of representations. In general, k-means clustering assumes a number of classes, k, and starts by assigning k centroids, c, to random samples in the dataset. In this case, k-means clustering operates on the set of representations, $h \in \mathbb{R}^d$, therefore each centroid will have the same dimensions as the representations, d. Let the matrix $C \in \mathbb{R}^{d \times k}$ represent the total set of centroids. Each sample can then be assigned to a cluster corresponding to which centroid is closest to its representation. Once every sample is labelled, the centroids themselves can be updated to the mean of all representations within that cluster. This process of i) assigning clusters and ii) updating centroids can be repeated until convergence or for a set number of iterations. This is illustrated in Eqn. 2.33, where the inner minimization selects the labels that give the shortest distance to a centroid, and the outer minimization assigns each centroid to the mean of its cluster.

$$\min_{C \in \mathbb{R}^{d \times k}} \frac{1}{n} \sum_{i=1}^{n} \min_{y_i \in \{0,1\}^k} \| f_\theta(x_i) - C y_i \|_2^2 \tag{2.33}$$

After k-means clustering is finished, the encoder network can be trained with a cross-entropy loss, treating the clusters as classes. Note that this stage trains the encoder network as well as the smaller fully connected network which has k outputs. The overall training procedure alternates between k-means clustering and cluster-based classification. Just as with the self-supervised method described above, the smaller fully connected network, g, can be discarded at the end of training and replaced with a linear classifier for the desired task.

Unlike the methods described thus far, autoencoders [Lecun, 1987, Ballard, 1987] define a loss in the input space. The network architecture typically involves an encoder, mapping inputs to a latent space, as well as a decoder, which maps latent codes back into images. The bottle neck of the network is normally chosen to be lower dimensional than the input space. An example is given in Figure 2.4.

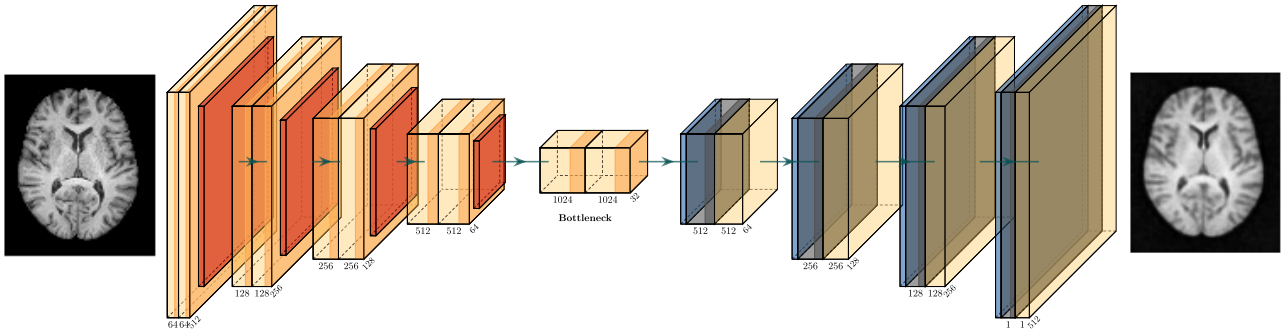


Figure 2.4: *Autoencoder trained to reconstruct images. Graphic produced with PlotNeuralNet [Iqbal, 2018].*

Autoencoders are trained to reconstruct inputs, usually with a mean squared error regression loss or similar loss function. A basic autoencoder loss is given in Eqn. 2.34, where f_θ represents the encoder and g_ϕ represents the decoder. Some autoencoders are also given corrupted inputs, $\tilde{x}$, and trained to reconstruct uncorrupted images [Vincent et al., 2010]. This denoising autoencoder loss, depicted in Eqn. 2.36, has parallels with the consistency loss in other methods described above. After training, autoencoders can be used to compress images into latent codes or reconstruct images based on the learned low dimensional representation of the data. The weights of the network can also be reused for training on a new task, discarding the decoder network.

$$\min_{\theta,\phi} \frac{1}{n} \sum_{i}^{n} \Big(x_i - g_\phi\big(f_\theta(x_i)\big) \Big)^2 \tag{2.34}$$

$$\tilde{x} \sim q_{\mathcal{D}}(\tilde{x}|x) \tag{2.35}$$

$$\min_{\theta,\phi} \frac{1}{n} \sum_{i}^{n} \Big(x_i - g_\phi\big(f_\theta(\tilde{x}_i)\big) \Big)^2 \tag{2.36}$$

One important difference from the previous unsupervised methods is that autoencoders are not trained on a discriminatory objective function. As such, the learned representation may not have the same qualities as the representations learned with SimCLR or DeepCluster. Encoders trained via SimCLR or DeepCluster can often be used for new tasks without changing any of the weight values. On the other hand, autoencoders are traditionally employed as a pre-training method; during the optimization for the new task, the weights continue to be updated as a way of fine tuning them. The representations learned from autoencoders may be less suitable for classification tasks, but they can have other advantageous qualities. For example, some representations can act as a low dimensional manifold of the data. Manipulating different values in this space can correspond to changes of specific features which can be visualized by reconstructing the image with the decoder.

In addition to the reconstruction loss, several autoencoders include a loss at the bottleneck. Variational autoencoders (VAE's) [Kingma and Welling, 2014, Rezende et al., 2014] are among the most well known in this category. Instead of encoding an image into a single vector, e.g. $\boldsymbol{h} \in \mathbb{R}^d$, VAE's map inputs to a set of d probability distributions (one for each dimension of the latent space). The standard VAE uses Gaussian distributions which are parameterized by a mean and standard deviation. As such, the encoder outputs two vectors $\boldsymbol{\mu}, \boldsymbol{\sigma} \in \mathbb{R}^d$. A single set of two vectors leads to d Gaussian distributions that can be sampled to obtain any number of regular latent codes. Using the decoder, these latent code samples can be mapped to images. Unfortunately this sampling process is problematic for backpropagation. As such, a reparameterization trick is used; the mean and standard deviation from the encoder are used to shift and scale a random sample from a unit Gaussian as shown in Eqn. 2.38 (note that $\odot$ refers

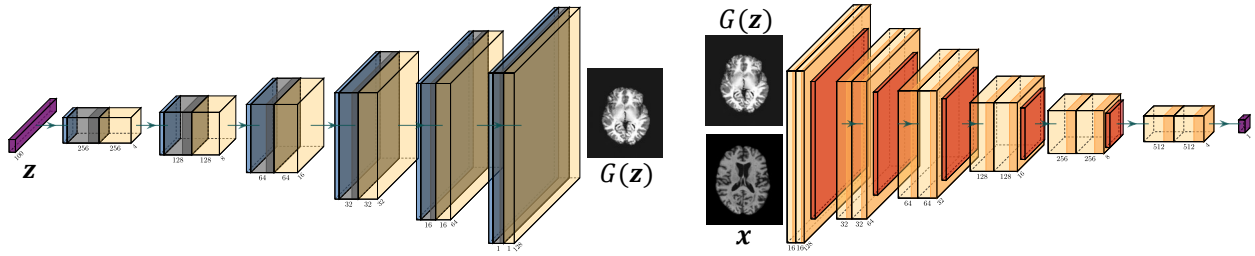
to the element-wise product). This arrangement makes it possible to compute the dependency on $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ and for gradients to flow from the decoder to the encoder. As mentioned earlier, there is also a loss applied directly to the encoder outputs. This loss regularizes the latent space by minimizing the KL divergence between the predicted distribution (parameterized by $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$) and a unit Gaussian prior. This is shown in Eqn. 2.39 and a complete derivation can be found in [Kingma and Welling, 2014]. Note that the order of the distributions in the KL divergence is reversed (compared to previously seen equations). This divergence is highest when $q(\boldsymbol{z})$ is high, but the prior, $p(\boldsymbol{z})$, is low. The regularization and design of the VAE latent space encourages the model to learn a representation with smooth transitions, where nearby points correspond to similar images.

$$\boldsymbol{\varepsilon}^{(l)} \sim \mathcal{N}(0, \mathbf{I}) \tag{2.37}$$

$$\boldsymbol{z}^{(i,l)} = \boldsymbol{\mu}^{(i)} + \boldsymbol{\sigma}^{(i)} \odot \boldsymbol{\varepsilon}^{(l)} \tag{2.38}$$

$$\begin{aligned} D_{KL}(q(\boldsymbol{z}) \| p(\boldsymbol{z})) &= \int q(\boldsymbol{z}) \big(\log q(\boldsymbol{z}) - \log p(\boldsymbol{z}) \big) d\boldsymbol{z} \\ &= -\frac{1}{2} \sum_{j=1}^{d} (1 + \log((\sigma_j)^2) - (\mu_j)^2 - (\sigma_j)^2) \end{aligned} \tag{2.39}$$

There are also alternative formulations of the bottleneck. For example, InfoVAE [Zhao et al., 2019] uses regular latent codes (the same as conventional autoencoders) and measures the divergence between these latent codes and a prior distribution using maximum mean discrepancy (MMD) [Gretton et al., 2007]. Using a kernel, $k(\boldsymbol{z}, \boldsymbol{z}')$, MMD projects samples into a higher dimensional space and computes the similarity between two samples. By computing the average similarity between all samples in a set, MMD can compare distributions. If two distributions are equal, then the average similarity between samples in each individual set will be equal to the average similarity between samples from the combined set. Another strategy is used in vector quantized methods (VQ-VAE's) [Van Den Oord et al., 2017, Razavi et al., 2019b]. These methods quantize latent codes into discrete entries in a codebook. Using discrete values simplifies the estimation of the probability that a given element in a latent code takes on a particular value. Several methods such as autoregressive models [Van Oord et al., 2016, Van den Oord



(a) Generator (b) Discriminator

Figure 2.5: GAN model trained to generate images and discriminate real images from generated ones. Graphic produced with PlotNeuralNet [Iqbal, 2018].

et al., 2016] can then be used to learn the distribution of the data based on the latent codes.

Another popular strategy in unsupervised learning is the use of generative adversarial networks (GAN's) [Goodfellow et al., 2014, Radford et al., 2016]. While autoencoders use pixel-wise regression, e.g. mean squared error, GAN's compare generated images and real images using a discriminator. The discriminator, D_ϕ, is simply an encoder trained with a cross-entropy loss; the labels are set to one for real images from the training data and zero for generated/fake images. The fake images are produced by the generator, G_θ, a decoder model which is given random latent codes as inputs. These latent codes are typically sampled from a prior distribution such as a uniform or Gaussian distribution. The gradients for the generator come from the discriminator. The two networks are optimized jointly based on Eqn. 2.40. The discriminator is optimized by maximizing Eqn. 2.40, which is equivalent to minimizing the binary cross-entropy of a standard classification task for real images (with label one) and fake images (with label zero). The generator is optimized by minimizing Eqn. 2.40, which leads to generating inputs that the discriminator classifies as real. Figure 2.5 shows an example generator and discriminator. Once trained, the latent space can be sampled to generate images. Just as with VAE's, the learned representation of a GAN can provide a low dimensional manifold of the data. Slowly varying elements in the latent codes can correspond to smooth transitions between generated images. In Figure 2.6, two random dimensions of the latent space are linearly changed within a fixed range.

$$\min_{\theta} \max_{\phi} \; \mathbb{E}_{\boldsymbol{x} \sim p_{data}} \Big[\log D_\phi(\boldsymbol{x}) \Big] + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}} \Big[\log \big(1 - D_\phi(G_\theta(\boldsymbol{z})) \big) \Big] \tag{2.40}$$

While GAN's can produce very realistic images, they often suffer from instability in training.

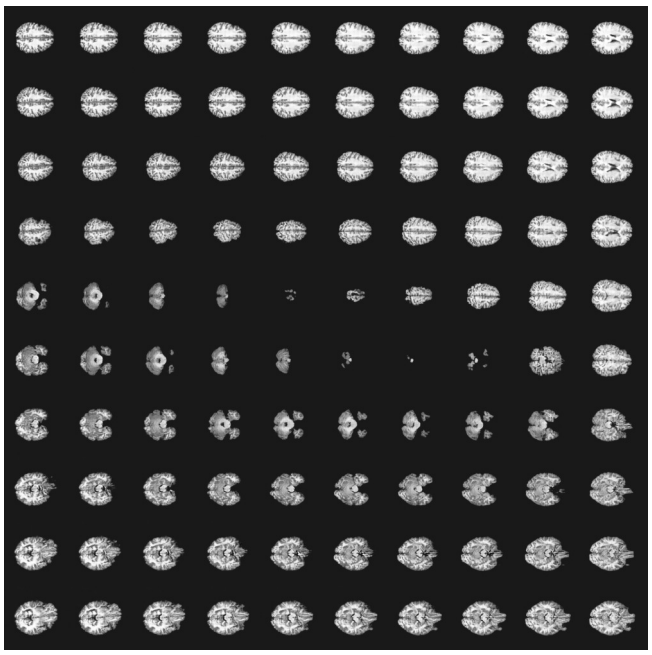


Figure 2.6: Interpolations in the latent space of a GAN correspond to smooth transitions between generated images.

Methods such as Wasserstein GAN's with gradient penalties [Arjovsky et al., 2017, Gulrajani et al., 2017] and GAN's with spectral normalization [Miyato et al., 2018a] impose different forms of regularization, particularly on the discriminator. Other approaches simplify the training process by generating smaller images at the start and progressively increasing the resolution throughout training [Karras et al., 2018]. One the most successful methods, known as StyleGAN [Karras et al., 2019, Karras et al., 2020], builds on top of the previously mentioned methods and redesigns the architecture of the generator. Instead of using a latent code as the input for the generator, StyleGAN injects the latent code at every layer of the generator using adaptive instance normalization (AdaIN) [Huang and Belongie, 2017]. For a given convolutional layer, AdaIN normalizes the activation values (layer outputs) to have zero mean and unit standard deviation; then it uses the latent code to scale and shift the activations values. This is shown in Eqn. 2.41, where $\boldsymbol{x}$ represents the activations and $\boldsymbol{y}$ represents parameters relating to the latent code. This technique has previously been used to alter the style of an image while preserving its content [Huang and Belongie, 2017]. Applying AdaIN at every layer in the generator allows the latent code to alter the style of the image at different resolutions, from coarse attributes to finer details. StyleGAN also inputs a source of stochastic variation

by injecting noise before each AdaIN layer.

$$\text{AdaIN}(\boldsymbol{x}_i, \boldsymbol{y}) = \boldsymbol{y}_{s,i}\frac{\boldsymbol{x}_i - \mu(\boldsymbol{x}_i)}{\sigma(\boldsymbol{x}_i)} + \boldsymbol{y}_{b,i} \tag{2.41}$$

This section has reviewed several unsupervised methods for learning representations that are useful for downstream tasks. At the core of each approach is an objective that is designed to extract useful features when performed repeatedly across the entire dataset. Among all of these methods, the coherence of the objective is an important consideration. Since there are no labels to enforce consistency, special care must be taken so that gradient steps on different batches (or tasks) are coordinated. If the objective is too unstable, the gradients can cause random jitter preventing the parameters from settling on a useful minima. The objective must also be sufficiently difficult such that the network cannot find a trivial solution. Depending on the objective function, the learned representation can have different qualities; this can make it more suitable for some tasks than others. As will be seen in later chapters, the utility of the learned features depends heavily on the overlap between the training objective and the target task.

2.4 Outlier Detection: Learning without Examples

Outlier detection is a unique setting, distinct from semi-supervised and unsupervised scenarios. A common assumption in outlier detection is that only normal data is available during training. This data is labelled, in the sense that it is known to be normal. However, the data is often provided as a single class without any subclass labels. This rules out the possibility of using a supervised discriminative objective function. It can also have implications for unsupervised methods, particularly those that learn to compare and contrast or cluster samples. These methods provide some of the best representations for downstream classification tasks, but they are often trained on multi-class data (withholding labels), sometimes with the same distribution as the test data. As seen in the previous sections, this can provide an implicit bias and good performance does not always translate to scenarios with a distribution mismatch [Oliver et al., 2018b]. The test data in outlier detection is, by definition, from a different data distribution.

Learning to detect outliers (without any examples of outliers to learn from) is a challenging and underdefined [Zhang et al., 2021] problem. Nevertheless, there are many practical applications including security/surveillance, quality control, content moderation, and disease screening. Below is an overview of some of the key strategies for outlier detection, with a focus on methods in the medical domain.

2.4.1 Reconstruction-based Approaches

A common strategy in outlier detection is to learn a model that can only reproduce normal samples. A given test sample can then be compared with the closest matching approximation that the model can produce. Ideally, the model will have all the features necessary to reconstruct the normal components perfectly, while lacking the features needed to reproduce abnormal content. Comparing the input to its reconstruction, e.g. with mean squared error, can thus provide an outlier score. Since the outlier score is computed in the input space, this naturally provides an interpretable localization of the deviating regions.

Many unsupervised methods can be used for reconstruction-based outlier detection. Autoencoders are a natural choice because their architecture is designed to project images onto a lower dimensional manifold and decode the result back to the input space. Baur et al. perform a comprehensive analysis comparing different types of autoencoders on MRI brain data [Baur et al., 2021]. Some types of autoencoders, such as VQ-VAE's, can be used to estimate the likelihood of elements in a latent code; components with a low likelihood can then be replaced with samples from the learned distribution [Marimont and Tarroni, 2021]. This method is used to restore unusual content and produce a more typical image. GAN's can also be used as reconstruction-based methods as long as there is a way to map images to the latent codes of the generator. One way to do this is to treat the latent code as a variable and optimize for the code which gives the lowest reconstruction error (while keeping the generator's parameters unchanged) [Schlegl et al., 2017]. Alternatively, an encoder can be trained specifically to encode samples, removing the need for optimization at test time [Schlegl et al., 2019]. Reconstruction-based methods can also be applied at different scales or resolutions to cater to different types of

outliers. Whole images give the most context and provide the complete anatomical structure. This can help with detecting global abnormalities or large defects and lesions. However, some generative models struggle to reproduce the finer details. For applications where texture is important, such as breast cancer screening, models can be trained at the patch level [Wei et al., 2018]. Using smaller patches can also simplify training for methods that struggle with stability at higher resolutions, e.g. GAN's [Alex et al., 2017].

Regardless of which model is used, there are several advantages and disadvantages that are inherent to all reconstruction-based methods. Part of the intuition behind reconstruction-based methods is that they should be able to detect any deviation because they reconstruct every feature in the image. In theory, this provides a more systematic sweep over all possibilities and circumvents the issue of specifying which features to focus on. However, in practice there is often an implicit bias based on the limitations of the model. If reconstructions are blurry, then fine grained anomalies could be missed and if the model only synthesizes textures, then it may not be capable of detecting structural anomalies. Another issue is the capacity of the model. Data with a wide range of natural variations may be harder to learn and any variations that are not captured by the model could be deemed anomalous (even if they are normal and present in the training data). Even if the model is able to produce high fidelity reconstructions, there are inherent limitations to computing outlier scores in image space. Image space distance does not correspond well to semantic differences. Trying to separate normal and abnormal samples in image space would be tantamount to classifying ImageNet [Deng et al., 2009] correctly without feature extraction. It can work in scenarios with well-behaved data and gross abnormalities, but subtle semantic differences can easily be hidden.

2.4.2 Embedding-based Measures

Another class of methods focuses on embedding samples into a meaningful latent space. Mappings from image space to latent space can be very flexible and expressive, allowing them to make subtle distinctions between classes [LeCun et al., 2015]. One method, named deep support vector data description (SVDD), trains an encoder to map normal samples to a compact

sphere ([Ruff et al., 2018]). Architectural constraints prevent the network from learning a trivial mapping of all samples to a single point. During testing, the sample is embedded in the latent space and the distance from the center of the sphere can be used as an outlier score. Another method compares patches in the latent space of an autoencoder [Alaverdyan et al., 2020]. In addition to the reconstruction loss, this method trains the encoder to maximize the cosine similarity between corresponding patches in brain data that has been normalized to the MNI template [Mazziotta et al., 2001].

Encoders are often trained to separate images of different classes, sometimes using labels. But in the case of outlier detection there are no labels and usually only one class is present in the training data. As such, the mapping learned by these embedding-based methods may not emphasize the most relevant features. The dimensions in the embedding space may correspond to feature variations exhibited within the training data. If the training data is composed of exclusively normal samples, then distances in the embedding space can be non-zero for completely normal images. If the training data is imperfect and contains a small percentage of outliers then the representation may learn these relevant features. Unfortunately, this representation may not be relevant for other types of *unseen* outliers. In general, embedding-based methods have the potential to learn semantic representations; but without any contrasting classes, it is not obvious what properties these representations will have.

2.4.3 Self-Supervised Tasks

Self-supervised methods have recently garnered attention in unsupervised applications. They have also been applied to outlier detection, particularly through self-supervised tasks involving geometric transformations [Golan and El-Yaniv, 2018]. Given a set of N transformations (e.g. rotation, translation, etc.), this method forms N artificial classes. A normal training image augmented using the i^{th} transformation is then associated with the i^{th} class. These transformed images and corresponding class labels are then used to train an encoder with a standard cross-entropy loss. This encoder will learn features that change with each transformation and ignore aspects that are unaffected by the transformations (i.e. features that are not informative).

As an example, consider horizontal flipping (left-right) as a transformation. If applied to chest X-ray data, the network may learn the orientation of asymmetrical structures such as the heart. Meanwhile, symmetrical structures such as the rib cage may be less helpful for the self-supervised classification task. At test time, all N transformations are applied to the test image and an outlier score is derived from the average classification accuracy on these augmented images. A low classification accuracy corresponds to a high outlier score because the image lacks the normal features that the network expects. In the X-ray example, a test image without a heart may result in a high outlier score because the network relies on learned asymmetrical structures to classify the orientation accurately. On the other hand, a missing or broken rib may be ignored because the ribs are not normally useful for identifying the horizontal flip transformation. Some methods add a second disjoint set of augmentations to create positive pairs (same image, different augmentation) and negative pairs (different images) for contrastive learning [Tack et al., 2020]. This helps to inject prior knowledge of class-preserving augmentations and can lead to better representations. It also allows for new outlier metrics such as the cosine similarity with the nearest (normal) training sample.

These methods perform much better than reconstruction or embedding based methods on datasets such as CIFAR-10 [Krizhevsky, 2009]. These types of datasets exhibit high intra-class variation. This can cause embedding-based methods to learn mostly normal variations which is less useful for detecting different classes. Image space distance also becomes less meaningful which puts reconstruction-based methods at a disadvantage. Although self-supervised methods excel in these challenging scenarios, they may be less suitable for medical anomalies. Most of the augmentations used in these methods encourage the network to detect the major structures in the image. But most patients, even those with irregularities, possess all of the major anatomical structures (organs, bones, etc.). Medical abnormalities can be subtle and highly semantic in nature. Most pathologies of interest can only be diagnosed by specialists with very specific domain expertise. As such, outlier detection in medical applications remains very much an open problem. The following chapters study these challenges in closer detail and propose solutions that are tailored to detecting medical outliers.

Chapter 3

Making the Most of Limited Labels

Publications Associated with this Chapter

[1] Tan, J., Au, A., Meng, Q., and Kainz, B. (2019). Semi-supervised learning of fetal anatomy from ultrasound. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pages 157–164. Springer

[2] Tan, J., Au, A., Meng, Q., FinesilverSmith, S., Simpson, J., Rueckert, D., Razavi, R., Day, T., Lloyd, D., and Kainz, B. (2020). Automated detection of congenital heart disease in fetal ultrasound screening. In *Medical Ultrasound, and Preterm, Perinatal and Paediatric Image Analysis*, pages 243–252. Springer

This chapter examines some of the challenges associated with medical data and studies different uses of labels within this context. Fetal ultrasound scans are exemplary for these purposes. As a modality, ultrasound is quite versatile and data acquisition is relatively easy. Unfortunately, this flexibility can lead to very unstructured data. For example, a large portion of the acquired data may not contain any diagnostically relevant information, but can create severe class imbalance. Another issue is that semantic distinctions in medical images can be based on obscure or subtle features. This can make classification more difficult without explicit, detailed annotations. This chapter explores i) whether semi-supervised methods can reduce annotation requirements and ii) how to maximize the utility of the labels that are available, through multi-task learning and test time perturbations. Although these methods cannot be used for outlier detection, where no pathology annotations are available, the experiments demonstrate that techniques

devised for natural images are often less effective in the medical domain. The same is true for outlier detection and the methods proposed in later chapters are tailored to address these subtle differences found in medical anomalies.

The chapter is organized into two sections based on publications listed above [Tan et al., 2019, Tan et al., 2020].

The first section considers the use of semi-supervised learning in standard plane detection. This is an important prerequisite step for nearly all applications in fetal ultrasound. An important finding in this work is that highly performant methods can face difficulty when applied to non-ideal data.

The second section presents a pipeline for automated detection of congenital heart disease. This pipeline uses supervised learning, multi-task learning, and adversarial perturbations at test time. The experiments demonstrate that diagnostic tasks can be difficult even when labelled examples are available. Using different sets of labels, with multi-task learning, can enhance the learned representation and improve the separation of healthy and pathological classes. The results of these experiments inform the design of the proposed methods in later chapters.

3.1 Semi-Supervised Learning of Fetal Anatomy from Ultrasound

3.1.1 Overview

Semi-supervised learning methods have achieved excellent performance on standard benchmark datasets using very few labelled images. Anatomy classification in fetal 2D ultrasound is an ideal problem setting to test whether these results translate to non-ideal data. Our results indicate that inclusion of a challenging background class can be detrimental and that semi-supervised learning mostly benefits classes that are already distinct, sometimes at the expense of more similar classes.

3.1.2 Introduction

Fetal ultrasound is the most widespread screening tool for congenital abnormalities and is a key recommendation in the World Health Organization's guidelines for antenatal care [World Health Organization et al., 2016]. Classification of standardized tomographic 2D planes is a key step in nearly all screening exams. However, low image quality and shortage of experts can compromise screening efficacy or in the least make quality heterogeneous across sites. To democratize care, several efforts have been made to automate standard plane detection using deep learning [Baumgartner et al., 2017, Cai et al., 2018, Chen et al., 2015a, Kong et al., 2018b]. However, many of these methods still rely on large amounts of labelled data.

Because labelling is expensive, semi-supervised learning (SSL) methods have become an active area of research, particularly for image data [Laine and Aila, 2017, Xie et al., 2019] and in the medical domain [Cheplygina et al., 2019]. Recent methods have achieved remarkable performance by learning from unlabelled data. This added information can help to push decision boundaries into lower density regions, resulting in better generalization. Amidst this progress, Oliver et al. call for more "realistic evaluation" [Oliver et al., 2018a]. One key concern is that benchmark datasets (e.g. CIFAR10, SVHN) do not reflect realistic scenarios.

We study the use of SSL for standard plane classification in fetal ultrasound [Baumgartner et al., 2017]. This task includes a challenging background class, class imbalance, and different levels of inter-class similarity. In accordance with Oliver et al. we demonstrate that supervised baselines can cope with surprisingly few labelled images; that a background class can cause SSL to become detrimental; and that SSL is effective for distinct classes, but can weaken performance on classes which are prone to confusion.

3.1.3 Related Work

Automatic anatomy detection from ultrasound videos is a popular topic with many successful approaches using convolutional neural networks [Baumgartner et al., 2017]. Further advancements have added multi-task learning to predict sonographer gaze [Cai et al., 2018] or multi-

scale networks to exploit lower and higher level features [Kong et al., 2018b]. Some methods have also studied the challenge of limited labelled data by pretraining on natural images [Chen et al., 2015a]. However there has been little investigation into the use of SSL. Exploring this avenue can reveal what benefits SSL can bring, and conversely, what challenges remain for SSL in non-ideal data scenarios.

Recently, SSL methods have gained momentum. Underlying many of these methods is a consistency loss which minimizes sensitivity to perturbations (in input/weight space). Examples of perturbations include image augmentations, gaussian noise, weight dropout [Laine and Aila, 2017], targeted augmentations [Xie et al., 2019], and mixup augmentations (interpolation between images) [Verma et al., 2019]. Input perturbations have been shown to minimize the input-output Jacobian (linked to better generalization) [Athiwaratkun et al., 2019]. Despite these advances, Oliver et al. point out that real unlabelled data is likely to be more irregular than the perfectly balanced benchmark datasets. It may also include out-of-distribution or confounding data that could hurt SSL [Singh et al., 2008]. We aim to study the ways in which SSL might help in a real problem which stands to benefit from SSL.

3.1.4 Methods

The architecture used in this study, Sononet [Baumgartner et al., 2017], is a convolutional neural network (similar to VGG [Simonyan and Zisserman, 2015]) that has been tailored for the task of anatomical standard plane detection in fetal ultrasound. It contains 15 convolutional layers, 4 maxpooling layers, and ends with global average pooling. This acts as a strong fully supervised baseline. Supervised methods are trained using Adam [Kingma and Ba, 2015] with a learning rate of 1E-3. All models are trained for 50 epochs with a batch size of 32.

For the SSL method, we use the consistency loss employed in both the Π model [Laine and Aila, 2017] and the unsupervised data augmentation (UDA) method [Xie et al., 2019] which are among the state of the art for standard benchmark datasets. This consistency loss uses the softmax predictions for unlabelled data, $\mathcal{D}_U$, as labels for the same images under augmentations. This consistency loss can be formalized as

$$\mathcal{L}_{\mathrm{KL}}(x_u, w) = KL(f(x_u; w)||f(x'_u; w)). \tag{3.1}$$

This is added to the typical supervised (cross-entropy) loss with some proportion λ. A value of $\lambda = 0.5$ is chosen as a heuristic and kept constant throughout all experiments. A higher λ would put more weight on unlabelled data, but many finer distinctions can only be learned through the labelled samples. As such, we specifically choose a λ that lets the supervised loss dominate. Note that the weights w are the same for inference on both the original image x_u (drawn from $\mathcal{D}_U$) and the corresponding augmented image x'_u. In this case the augmentations include random combinations of the following operations:

- Horizontal flipping
- Random contrast adjustment by a factor within [0.7, 1.3]
- Random rotation by an angle within $[-\frac{\pi}{4}, \frac{\pi}{4}]$
- Random cropping ranging from 1% to 20%

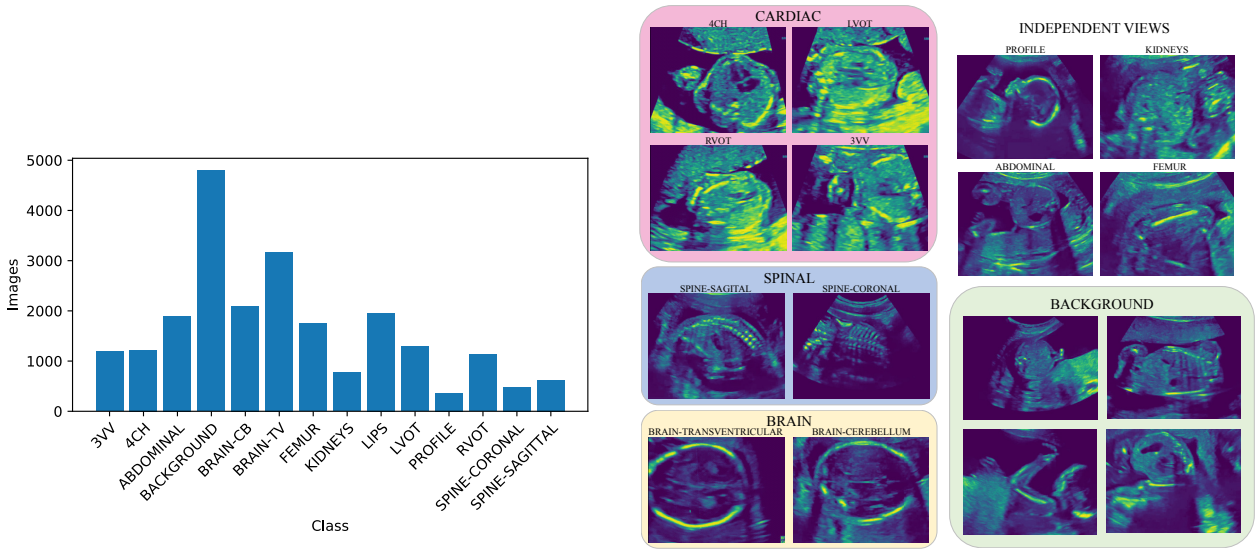
The same augmentation is applied to all methods, including supervised baselines. These augmentations are not specially tailored to the data as is the case in UDA. For the best performance, UDA uses AutoAugment [Cubuk et al., 2019], a reinforcement learning method that finds the optimal augmentation policies. This would likely improve both SSL and fully supervised regimes. However, as shown in the results, the scores of the supervised baselines are already very high with surprisingly few labelled images; applying AutoAugment to both SSL and fully supervised methods may truncate the margin for potential improvement that we wish to study. We include a) training signal annealing (TSA), b) confidence-based masking (CBM), c) entropy minimization, and d) softmax temperature controlling [Xie et al., 2019]. Since these are proposed in [Xie et al., 2019], we refer to the combination of i) the consistency loss with ii) these additional techniques, as UDA for simplicity and to acknowledge their contributions.

TSA [Xie et al., 2019] uses a threshold, η_{tsa}, to mask the contribution of a given labelled image, x_l (drawn from $\mathcal{D}_L$), to the supervised gradient. Specifically, an example only contributes to

the gradient if the softmax probability in the ground truth class is greater than the threshold, $p(y^*|x_l) > \eta_{tsa}$. The threshold η_{tsa} starts at $\frac{1}{\#ofclasses}$ and is increased to 1 following a linear, log, or exponential schedule across the total number of epochs.

CBM [Xie et al., 2019] uses a confidence threshold on unlabelled images. An unlabelled image only contributes to the consistency gradient if, $\max(p(y|x_u)) > \eta_{cbm}$. A η_{cbm} of 0.75 is used for all SSL experiments.

Entropy minimization [Grandvalet and Bengio, 2005] is applied to $p(y|x'_u)$, the prediction for an augmented unlabelled image. Also the prediction for original unlabelled image is sharpened by using a softmax temperature of 0.8.



(a) *Data Distribution* (b) *Standard Plane Views*

Figure 3.1: *Distribution of the training data (a) and examples of standard views (b).*

Dataset

Our dataset contains 13 anatomical classes plus 1 background class. The entire training dataset consists of 22757 images (class distribution shown in Figure 3.1.a). For SSL, 100 images are extracted from each class to make up the labelled data $\mathcal{D}_L$. The remaining images are treated as unlabelled data $\mathcal{D}_U$. Experiments are performed using subsets of $\mathcal{D}_L$, specifically using 1, 5, 20, 50, and 100 images per class. The test set follows the same distribution, but totals to

5737 images. Each image is 224x288 pixels. These image frames are derived from a dataset containing 2438 videos from over 2000 volunteers.

Examples of some of the classes are displayed in Figure 3.1.b. The cardiac classes are among the most difficult to distinguish. While the spine and brain also span multiple classes, these images are generally more clear and can have significant pose differences (spine). The background class contains a diverse range of images sampled from the videos (excluding frames of standard planes). All extracted samples satisfy a minimum image-space distance between neighbouring frames. This means that most background frames are extracted during rapid probe movement and not when the sonographer slows down to home in on the standard planes. However, some images that resemble standard planes still make it through this naïve filtering approach. Also, the background class is larger than any anatomical class (Figure 3.1.a). In short, the background class introduces class imbalance; is not easily characterized by a single common feature; and contains examples which resemble other classes.

Evaluation

Fully supervised methods are evaluated with 1, 5, 20, 50, and 100 labelled images per class. Another experiment is performed using the entire labelled training set (total 22757 images). The SSL framework is applied to the cases of 5, 20, and 50 labelled images per class, where there is a large margin for potential improvement. These cases represent a very feasible labelling task compared to labelling all 22757 images.

Each evaluation is done with and without the background class. To measure the impact of the background class on anatomical classes, accuracy is reported only for the anatomical classes (not background). Accuracy is also reported with a single merged cardiac class. In this case, any cardiac view that is classified as any of the four cardiac views is considered correct. Comparing the overall and grouped cardiac accuracies gives an indication of whether improvements extend to the cardiac classes.

3.1.5 Results

Accuracy for fully supervised baseline methods is shown in Figure 3.2. With only 20 labelled examples per class, overall accuracy is near 70% and grouped cardiac accuracy is over 80%. All baselines are trained for 50 epochs and did not grossly overfit (except for 1 image per class which was trained for 5 epochs).

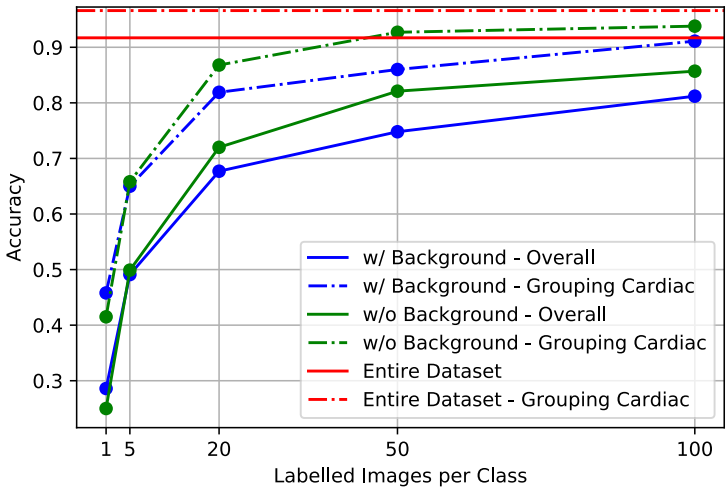


Figure 3.2: *Supervised baselines with varying number of labelled images per class. Inclusion of the background class increases the difficulty of the classification task. Accuracy is reported as either overall or grouping all cardiac classes into a single class.*

When applying the SSL consistency loss, we find the performance is sensitive to the consistency loss masking threshold, η_{cbm}. This is particularly true for the cardiac classes. Table 3.1 compares the performance of different thresholds for the cardiac classes. The threshold for all other classes remains constant at 0.75. A cardiac threshold of 0.25 gives best performance and is used for all further experiments.

Further experiments are performed with different UDA settings to find the optimal configuration (Table 3.2). We find that a log TSA schedule gives the best performance. Log schedules are suggested for cases when the network is less likely to quickly overfit[Xie et al., 2019].

The best found configuration is then used with 5, 20, and 50 labelled images per class, with and without the background class. Accuracies are reported in Figure 3.3.

Confusion matrices are displayed in Figure 3.4. SSL improves accuracy for distinct classes

Table 3.1: Confidence mask threshold values for cardiac classes. These experiments use 20 labelled images per class and exclude the background class.

Method	Cardiac Confidence Mask Threshold	Grouping Cardiac	Overall
Supervised	N/A	**0.868**	**0.720**
Basic UDA	η_{cbm}	0.849	0.665
	$\frac{1}{2} \cdot \eta_{cbm}$	0.840	0.661
	$\frac{1}{3} \cdot \eta_{cbm}$	**0.905**	**0.728**
	$\frac{1}{4} \cdot \eta_{cbm}$	0.831	0.664
	> 1 (disabled)	0.631	0.631
UDA Best Configuration	η_{cbm}	0.915	0.720
	$\frac{1}{3} \cdot \eta_{cbm}$	**0.936**	**0.754**

Table 3.2: Different configurations for the case of 20 labelled images per class without the background class.

Method	Optimizer	TSA Schedule	Grouping Cardiac	Overall
Supervised	Adam: 1E-3	N/A	**0.868**	**0.720**
UDA	Adam: 1E-3	Linear	0.905	0.728
	Momentum: 1E-3	Linear	0.891	0.692
	SGD cyclic:[7E-3,5E-2]	Linear	0.921	0.735
	Adam: 1E-3	Log	**0.936**	**0.754**
	SGD cyclic:[7E-3,5E-2]	Log	0.935	0.744

such as brain, femur, kidney, and lips from mid 0.80 (a - supervised) to mid 0.90 (b - UDA), which approaches the fully supervised performance shown in (c). However, for cardiac classes, confusion is increased when using SSL.

3.1.6 Discussion

Similarly to Oliver et al. [Oliver et al., 2018a], we show that supervised baselines are surprisingly accurate (Figure 3.2). There is also a clear diminishing return of increasing the number of labelled images, which starts as early as 20 examples per class. Inclusion of the background class decreases accuracy in almost all cases. This indicates that the background class adds difficulty even in the fully supervised case.

Investigating sensitivity to confidence thresholds (Table 3.1), we see that $\frac{1}{3} \cdot \eta_{cbm}$ (0.25) is the only setting that improves both grouped cardiac and overall accuracy for the basic UDA imple-

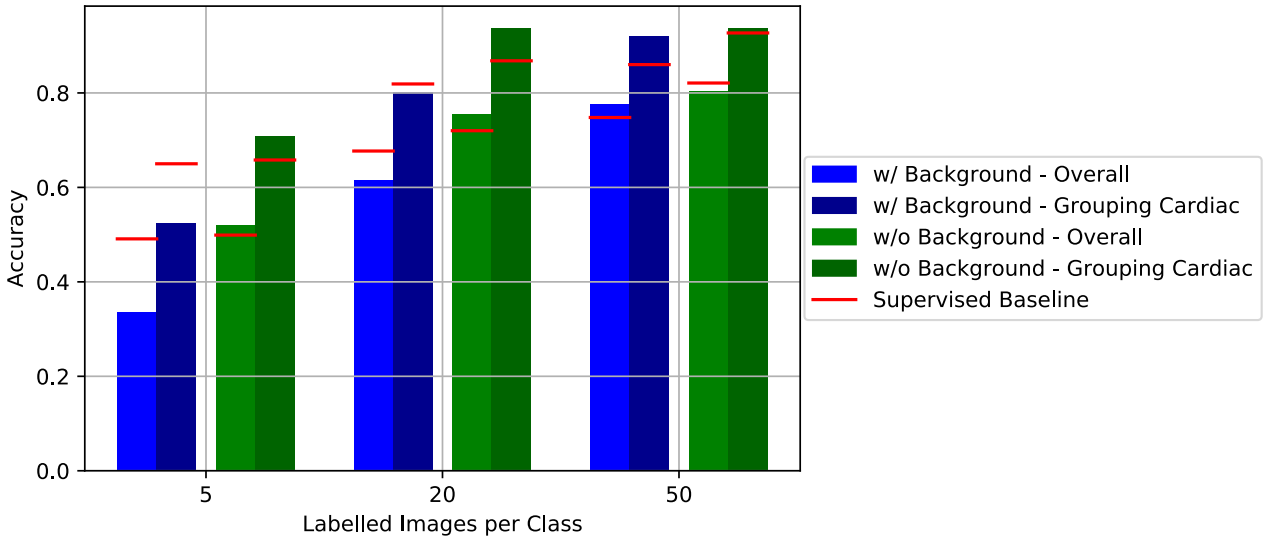
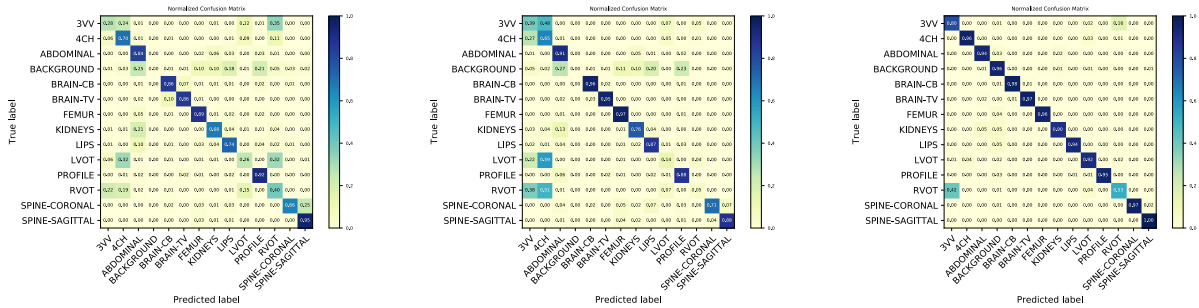


Figure 3.3: *Overall and grouped cardiac accuracies with and without the background class for 5, 20, and 50 labelled images per class. Red bars indicate supervised baseline performance. Without the background class (green bars), SSL accuracy is almost always above supervised baslines (red). Inclusion of the background class can cause SSL performance to drop below supervised baselines.*

mentation. A value near 0.25 is reasonable given that the network must divide its confidence over 4 very similar cardiac classes. Even for the best UDA configuration, a threshold of 0.25 for the cardiac classes makes a considerable difference; without it, overall accuracy does not improve from the supervised baseline.

Figure 3.3 displays that without the background class (green bars), the SSL regime can almost always improve upon the fully supervised baseline. The only exception being the overall accuracy for 50 labelled examples. In this case the supervised baseline is already quite high and any further improvement would likely depend on the cardiac classes which the SSL method struggles with. In contrast, the inclusion of the background class (blue bars), not only reduces the accuracy of the fully supervised baselines, but tends to be harmful to the SSL method. For most of the blue bars, the SSL method fails to match, let alone surpass, the baseline accuracy. This illustrates the negative impact of including confounding images in the unlabelled data. Again, the case with 50 labelled examples is the exception. Perhaps 50 labelled examples is sufficient to capture the majority of the variation in the background class, preventing it from having a negative impact on the SSL loss.



(a) *Supervised baseline* (b) *UDA* (c) *Supervised on all data*

Figure 3.4: *Confusion matrices for the case of 20 labelled images per class without background. The supervised baseline (a) performs surprisingly well given the limited data. SSL (b) is able to make considerable improvement for distinct classes, but can increase confusion of cardiac classes. Even when trained on the entire labelled dataset (c), some cardiac classes are prone to confusion.*

While the SSL has been shown to increase both grouped cardiac and overall accuracies, the confusion matrices in Figure 3.4 clearly show that these improvements are often at the expense of the cardiac classes. It seems unlabelled data can help the network to learn when the classes are inherently more distinct, but can cause harm when classes are inherently similar.

3.1.7 Summary

Supervised baselines provide surprisingly reliable performance even in extremely low data regimes (e.g. accuracy of 50% from only 5 labelled images per class). Recent developments in SSL can further improve this performance. However, irregular data can cause SSL to be detrimental rather than beneficial. Also, classes with high similarity, such as cardiac views, can see an increase in confusion. For such classes, injecting domain knowledge (e.g. lowering cardiac confidence thresholds) may be necessary to supplement a lack of labelled examples.

3.2 Automated Detection of Congenital Heart Disease in Fetal Ultrasound Screening

3.2.1 Overview

While the previous section dealt with standard plane detection, this section progresses to the actual diagnostic task of detecting congenital heart disease. For some cardiac abnormalities, prenatal screening with ultrasound can lower neonatal mortality substantially. However, the need for human expertise, coupled with the high volume of screening cases, limits the practically achievable detection rates. This section explores the potential for deep learning techniques to aid in the detection of congenital heart disease (CHD) in fetal ultrasound. We propose a pipeline for automated data curation and classification. During both training and inference, we exploit an auxiliary view classification task to bias features toward relevant cardiac structures. This bias helps to improve in F1-scores from 0.72 and 0.77 to 0.87 and 0.85 for healthy and CHD classes respectively.

3.2.2 Introduction

Ultrasound is the foremost modality for fetal screening. Its portability, low cost, and fast imaging make it one of the easiest imaging modalities to deploy. This gives front-line sonographers the tools to perform screening at a *population* level. In each fetal examination, sonographers must inspect a wide range of anatomical features including but not limited to the spine, brain, and heart. The breadth of this task makes it difficult for sonographers to develop specialized expertise for every anatomical feature. Unfortunately, this can lead to some conditions going undiagnosed. The most fatal of which is congenital heart disease (CHD) which is associated with over 47% of perinatal deaths and over 35% of infant deaths (only considering deaths related to congenital abnormalities) [National Congenital Anomaly and Rare Disease Registration Service, 2017].

Experts can detect CHD's with over 98% sensitivity and near 90% specificity [Bennasar et al.,

2010, Yeo et al., 2018]. However, shortage of specialists means that over 96% of examinations are performed by generalist sonographers [Van Velzen et al., 2016]. As a result, *population*-based studies consistently report detection rates around 39% [Pinto et al., 2012] (with one exception reaching 59% [Van Velzen et al., 2016]). Machine learning methods could help close this gap and provide sonographers with assistance for more difficult diagnoses.

Deep learning has been used in many fetal ultrasound applications including standard plane detection [Baumgartner et al., 2017, Cai et al., 2018, Chen et al., 2015b, Kong et al., 2018a], extrapolation of 3D structure from 2D images [Cerrolaza et al., 2018], and biometric measurements for developmental assessment [Sinclair et al., 2018, Kim et al., 2018]. However, there have been relatively few works on diagnostic assistance in fetal screening. Some of the major challenges in this application are i) data curation and ii) disease variance. Data curation is crucial because only certain "standard planes" are considered diagnostic [National Health Service and Public Health England, 2018]. Extracting relevant frames for CHD is particularly challenging because pathological cases *by definition* deviate from the description of these standard planes. The high variation of the manifestation of CHD's also make diagnosis difficult.

In this work we propose a pipeline to perform automated diagnosis of hypoplastic left heart syndrome (HLHS), a term which encompasses a spectrum of malformations in the left ventricle and its outflow tract [Simpson, 2000]. A standard plane detector is first used to extract relevant frames from the healthy and pathological cases. This data is then used to train a classifier to distinguish between normal control (NC) cases and HLHS patients.

3.2.3 Related Work

Most successful applications of deep learning in fetal ultrasound have been in pre-diagnostic tasks. In particular, standard plane detection has been studied extensively[Baumgartner et al., 2017, Cai et al., 2018, Chen et al., 2015b, Kong et al., 2018a]. Standard plane detection is also involved in the proposed pipeline and is based on the SonoNet architecture [Baumgartner et al., 2017].

Most closely related to our work is a recent study on diagnosing HLHS and Tetralogy of Fallot (TOF) in fetal ultrasound [Arnaout et al., 2018]. In their methodology, images are labelled based on standard planes allowing for extraction of relevant images with high diagnostic quality. Using these images, they train a series of binary classifiers to distinguish between healthy and pathological cases. Each classifier is trained using images from only one standard plane. The predictions from each plane are then summed and a threshold is used to determine the final diagnosis.

They achieve high sensitivity and specificity demonstrating that neural networks can learn to identify CHD. In their study, the training images are of high diagnostic quality and come from a diverse dataset including about 600 patients (for normal vs. HLHS). These favorable conditions are not always possible because of a lack of expert annotations and the rarity of CHD conditions. As such, we aim to investigate the feasibility of CHD detection in the low data regime, using a total of 100 patients. We also study the impact of automated and manual data curation.

The design of [Arnaout et al., 2018] is also dependent on data curation. Training individual networks for each standard plane means that training data must be accurately sorted into the correct planes. It also means that test data must be accurately sorted for inference because each network is only trained to recognize pathology in a single view. Furthermore, using individual networks for each plane leads to larger memory requirements. Instead, we train a single network for all views and explore the use of plane labels as an auxiliary task for multitask learning [Caruana, 1997].

3.2.4 Method

The proposed pipeline consists of two key stages, plane extraction and pathology classification. Given uncurated data, this automated workflow allows for a model to be trained for CHD classification. Figure 3.5 depicts the overall workflow.

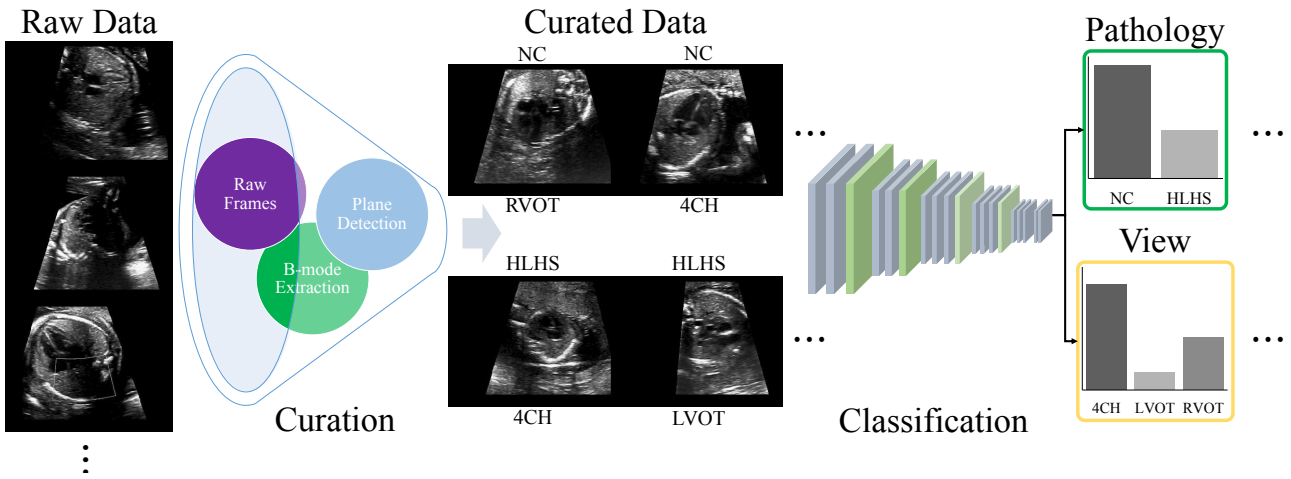


Figure 3.5: *Automated curation and classification pipeline.*

Data Characteristics

Data was collected from 39 healthy patients and 61 patients diagnosed with HLHS, for a total of 100 patients. Each patient's data is collected as raw ultrasound DICOM videos. These videos are unannotated, meaning they contain i) frames of irrelevant anatomy, ii) frames in the vicinity of the heart, and iii) precise standard planes which are typically used for diagnosis. Table 3.3 summarizes the distribution of frames. The final extracted frames used for training are greyscale images of dimensions 224x188 pixels. The curation process used to go from raw videos to extracted frames is described in the following section.

Table 3.3: Distribution of patients, clean cardiac frames, and different cardiac views.

	NC			**HLHS**		
Unique Patients	39			61		
Total frames in DICOM files	354867			741290		
Clean frames after curation	189000			376337		
Clean frames (cardiac views)	102993			143468		
View	4CH	LVOT	RVOT	4CH	LVOT	RVOT
Extracted Frames	31938	62195	8452	49176	73203	19959

Plane Extraction

The data curation pipeline involves i) extraction of B-mode frames, and ii) standard plane extraction. In the B-mode extraction phase, frames are run through a series of tests which detect the presence of certain colors, user interface elements, or particular histogram characteristics. These help remove Doppler, split-view, and M-mode frames respectively.

A standard plane detector, SonoNet [Baumgartner et al., 2017], is used to extract relevant cardiac frames. Specifically, the 4 chamber heart (4CH), left ventricular outflow tract (LVOT) and right ventricular outflow tract (RVOT) views are used. Table 3.3 displays the number of frames extracted for each view. Note that this standard plane detector has been trained to detect cardiac frames in *healthy* patients. As such, its ability to detect relevant frames in patients with malformed hearts may be impaired. Given the bias toward normal hearts, the detector may not recognize frames which show gross defects, which would be the most diagnostically relevant. Instead it may favor instances where the heart appears closer to normal, potentially making them more difficult to distinguish. While this is unideal, it represents the most general case where we do not have access to annotations which highlight the most diagnostic frames. This circumvents the need to train individual plane detectors for every pathology. To test whether diagnostic information can be gleaned from these images, we train a classifier as described in the following section.

CHD Classification

We train a single classifier to discriminate between healthy and HLHS patients. All three cardiac views are used for training within the same network. The network architecture is the same as SonoNet [Baumgartner et al., 2017] which has been inspired by the staple VGG network [Simonyan and Zisserman, 2015]. A standard binary cross-entropy loss (Eqn. 3.2) is used for optimization.

In the low data regime (particularly when the number of unique patients is low) it is difficult to ensure that the classifier learns features that are genuinely related to pathology. Instead,

the network might learn extraneous features that are reliable within the training data but are not robust within the test set. It is also prohibitively expensive to annotate which regions are important in each image. As such, we exploit the view labels which come for free from the data curation pipeline. Discriminating between the 4CH, LVOT and RVOT views requires the network to identify cardiac structures. It is also an intra-patient task, meaning that patient-specific features are not reliable. View classification is thus added as auxiliary task to bias the network toward cardiac structures and away from memorization of patient-specific characteristics that do not generalize to test data. The auxiliary loss and combined multitask [Caruana, 1997] loss are given in Eqn. 3.3 and Eqn. 3.4 respectively.

$$\mathcal{L}_{\text{CHD}}(x_i, y_i, f) = -\sum_{c=1}^{N=2} y_{i,c} \log(f(x_{i,c})) \tag{3.2}$$

$$\mathcal{L}_{\text{view}}(x_i, v_i, f) = -\sum_{c=1}^{N=3} v_{i,c} \log(f(x_{i,c})) \tag{3.3}$$

$$\mathcal{L}_{\text{Multitask}}(x_i, y_i, v_i, f) = \mathcal{L}_{\text{CHD}} + \lambda \mathcal{L}_{\text{view}} \tag{3.4}$$

$$\lambda_i = 1 - 0.5\mathcal{L}'_{\text{CHD}}(x_i, y_i, f) \tag{3.5}$$

The weight of the view loss, λ, is computed individually for each instance (Eqn. 3.5). It is based on the minmax-scaled CHD loss values, $\mathcal{L}'_{\text{CHD}} \in [0, 1]$, of each sample within a batch. The view loss only increases when a sample has a lower CHD loss, making CHD the network's priority. There are more sophisticated ways of finding the optimal weighting between tasks, e.g. [Kendall et al., 2018], but this simplistic approach has little overhead and demonstrates an improvement over a naive setting of $\lambda = 1$.

Assessing Diagnostic Quality during Inference

The automated curation pipeline helps to reduce manual data processing. However, it may extract low quality frames which have less diagnostic information. Including predictions on such frames adds noise to the overall diagnosis. As such, we aim to identify and remove unreliable predictions. Note that even if *all* frames from a patient are rejected, it is better than

providing a diagnosis which is unreliable.

A reliable prediction for cardiac disease should depend on the cardiac structures themselves, rather than extraneous features in the background. Such a prediction should be robust to small perturbations as long as the cardiac structures are kept intact. Our aim is to generate these perturbations and use them to identify which predictions are reliable.

Generating perturbations that preserve the cardiac structures requires knowledge of the location of the cardiac structures. However, annotating these structures is expensive. As such, we exploit the auxiliary view task as a proxy. The view classification task determines which cardiac plane is presented in the image, which depends heavily on the cardiac structures. To generate a perturbation we use the gradient from the view task to distort the image in a way that does not change the the view prediction (i.e. the cardiac structures). This is similar to adversarial examples [Szegedy et al., 2014], except we keep the prediction the same (Eqn. 3.6) by following the negative gradient (Eqn. 3.7 [Madry et al., 2018]). Multiple perturbation steps, δ, can be taken; each time using a step size of $\alpha = 2$ up to a maximum perturbation of ± 8 from a pixel's original value $p \in [0, 300]$.

$$\min_{\delta \in \Delta} \mathcal{L}_{\text{view}}(x_i + \delta, v_i^*, f) \ , \ \text{where } v_i^* = f(x_i) \tag{3.6}$$

$$\delta = -\alpha \text{sgn}(\nabla_x \mathcal{L}_{\text{view}}(x_i, v_i^*, f)) \tag{3.7}$$

Evaluation

After automatically extracting the relevant frames for all patients, *manual* curation was performed on images from roughly 20% of the patients. In manual curation, the images are sorted into three groups based on their diagnostic quality (high, medium, and low) by expert clinicians. Half of these patients are used for testing and the other half are used for validation. Evaluation on the test data measures how well the model is able to learn generalizable features from the automatically curated data. The quality of the automated curation can also be measured by comparing results on test sets with and without manual curation. We report precision, recall, F1-scores and the area under curve for the receiver operator characteristic curve (ROC-AUC).

3.2.5 Results

An overview of the results is presented in Table 3.4. The test data produced by automated curation contains images with varying levels of diagnostic quality. Testing on all levels (low-high quality) results in poor performance. Using medium-high quality immediately improves all metrics. The multitask loss provides a considerable improvement and the weighted multitask approach ($\lambda \propto \mathcal{L}_{\text{CHD}}$, Eqn. 3.5) further improves performance. Using robust inference (described in Section 3.2.4) helps to remove predictions that are deemed less reliable. The fact that performance increases indicates that robust inference is effective in identifying and excluding predictions that are less accurate.

Table 3.4: Pathology classification results for different models using testing data with different diagnostic quality.

Loss	**Test Quality**	**Precision (NC:CHD)**		**Recall (NC:CHD)**		**F1-score (NC:CHD)**		**ROC-AUC**
CHD	All	0.72	0.64	0.59	0.76	0.65	0.70	0.75
CHD	Med-High	0.72	0.77	0.71	0.77	0.72	0.77	0.82
Multitask ($\lambda = 1$)	Med-High	0.77	0.81	0.77	0.81	0.77	0.81	0.87
Multitask ($\lambda \propto \mathcal{L}_{\text{CHD}}$)	Med-High	0.80	0.80	0.74	0.85	0.77	0.83	0.89
Multitask ($\lambda \propto \mathcal{L}_{\text{CHD}}$)	Robust Frames	0.83	0.89	0.90	0.82	0.87	0.85	0.93

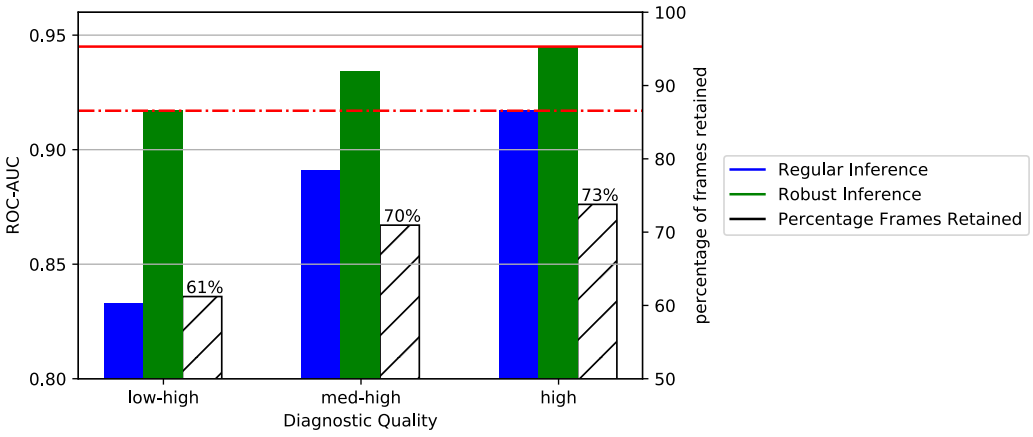


Figure 3.6: *ROC-AUC comparison of regular and robust inference. Including low quality frames quickly leads to a sharp decline in performance (blue). Robust inference can help to identify frames that are more reliable (white hatch, right y-axis). Evaluating only reliable frames leads to more accurate overall predictions (green).*

A closer examination of robust inference is given in Figure 3.6. Using regular inference (blue), the ROC-AUC becomes severely compromised when including low quality images. However,

the performance can be recovered by using robust inference (green) to automatically determine when a prediction is less reliable.

The white hatched bars (Figure 3.6) indicate the percentage of images which are deemed reliable in each set. The percentage of reliable images increases as the expert-rated quality increases. This indicates that the proposed reliability rating corresponds to expert judgement to some extent. Ideally 100% of the high quality images should be retained; however 73% are deemed reliable. Nonetheless the excluded predictions were indeed unreliable and their exclusion results in an improvement of almost 0.03 (solid and dotted red lines). In the low-high set, only 61% of frames are retained. To put this in perspective, experts only considered 39% of the low-high set as being of acceptable quality (medium or above).

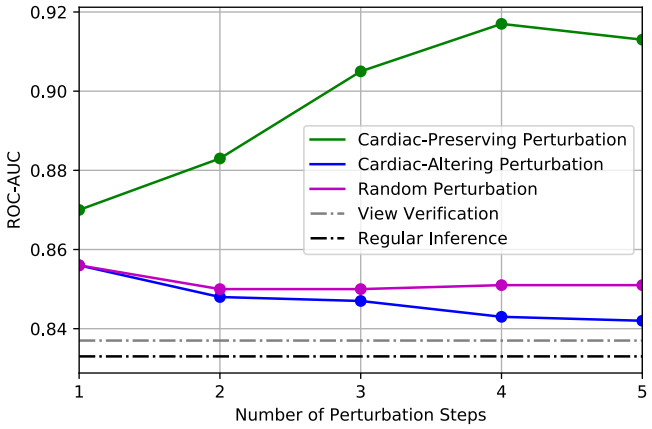


Figure 3.7: *ROC-AUC using different approaches of frame quality assessment. All methods improve upon regular inference (black). Cardiac-preserving perturbations stand out as being considerably better at removing unreliable predictions (green).*

We also compare different approaches to robust inference in Figure 3.7. The view verification approach excludes any images which produce an incorrect view prediction (based on 'ground truth' view labels provided by the plane detector in the curation pipeline). This gives very limited improvement (grey).

The proposed cardiac-preserving perturbations are able to find and remove inaccurate predictions, which consistently improves ROC-AUC (Figure 3.7, green). In fact, increasing the number of perturbation steps actually leads to better results. In comparison, random (purple) and cardiac-altering perturbations (i.e. adversarial examples - blue) do not provide as much improvement.

3.2.6 Discussion

Various elements, including multitask learning [Caruana, 1997] and robust inference, contribute toward improving the ROC-AUC from 0.75 to over 0.91. With a small number of unique patients and a lack of manual curation, the standard classifier performs poorly. Multitask learning [Caruana, 1997] helps to provide a bias toward cardiac structures that are relevant for the view classification task. These features are also helpful for pathology classification and lead to an increase in performance (Table 3.4).

Robust inference also helps to improve the scores by filtering out unreliable predictions. Predictions are considered less reliable if they change when the image is perturbed in a way that preserves cardiac features. Stronger perturbations (using more steps) are more likely to alter predictions. However, predictions that rely on cardiac features should remain unaffected. Figure 3.7 demonstrates that strong cardiac-preserving perturbations are the most adept at finding unreliable predictions. In comparison, random and cardiac-altering (adversarial) perturbations are not able to tease apart reliable and unreliable predictions.

We find that robust inference is an important component because the automated curation pipeline includes many images of low diagnostic quality. Typically, these frames would have to be manually vetted in order to reduce the noise in the overall prediction. However, robust inference can serve as a filter to remove predictions from low quality images. For low-high quality images, robust inference improves the ROC-AUC from 0.83 to 0.92. This is on par with the ROC-AUC achieved with regular inference on *only* high quality images (0.92). This is highlighted in Figure 3.6 with a red dotted line.

Measuring the reliability of a prediction is similar to uncertainty estimation (e.g. [Gal and Ghahramani, 2016]). However, uncertainty quantification is a much more complex task and is not always straightforward to interpret [Yao et al., 2019]. Instead, we simply measure the prediction's dependence on cardiac structures (which should hold key diagnostic information). With this prior knowledge we can estimate *our* confidence in the prediction, rather than the network's intrinsic uncertainty.

We demonstrate that it is possible to train a classifier to identify CHD from standard B-mode images. In the future, temporal information or Doppler images could be used to provide more diagnostic information. Nevertheless, this work represents a step toward the goal of assisting front-line sonographers with CHD diagnosis at a population level.

3.2.7 Summary

This section proposes an automated pipeline for curating and classifying CHD in fetal ultrasound. The curation step extracts relevant frames which makes training possible. Still, automated curation includes many images with poor diagnostic quality. Noisy predictions on such images can lead to a reduction in overall performance. As such, robust inference is introduced in this work to exclude predictions which may be less trustworthy. This helps to achieve a F1-score of 0.87 and 0.85 for NC and CHD classes respectively and an ROC-AUC of 0.93.

Overall this chapter demonstrates that medical data can exhibit non-ideal characteristics and class distinctions can be difficult to learn without detailed annotations. To tackle these problems, it is often necessary to adapt conventional methods, as seen throughout this chapter. But some problems call for new alternative strategies. Outlier detection can be considered as an extreme case of class imbalance, where only one class is available. Medical applications can be particularly difficult because anomalous classes can be subtle and difficult to detect even with supervised learning. The following chapters present methods that aim to learn subtle distinctions while using data from only one class.

Chapter 4

Divergent Search for Few-shot Image Classification

Publications Associated with this Chapter

[1] Tan, J. and Kainz, B. (2020). Divergent search for image classification behaviors. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 91–92. https://doi.org/10.1145/3377929.3389973

The previous chapter showed that existing methods for semi-supervised learning are only helpful when the unlabelled data is evenly distributed among equally distinct classes. This chapter deals with strategies for unsupervised learning that are less dependent on the compatibility between the unlabelled (training) data and the target task. Afterall, the goal in outlier detection is to detect unseen classes, so the exact target task is generally unknown prior to testing. The proposed method in this chapter uses divergent search to accumulate a repertoire of features [Tan and Kainz, 2020]. This approach is inspired by the way that evolution diversifies in order to fill all of the niches in a fitness landscape [Ricklefs, 2010]. Once this repertoire has been mined from the training data, it can be harnessed for the target task through few-shot learning.

4.1 Overview

When data is unlabelled and the target task is not known a priori, divergent search offers a strategy for learning a wide range of skills. Having such a repertoire allows a system to adapt to new, unforeseen tasks. Unlabelled image data is plentiful, but it is not always known which features will be required for downstream tasks. We propose a method for divergent search in the few-shot image classification setting and evaluate with Omniglot and Mini-ImageNet. This high-dimensional behavior space includes all possible ways of partitioning the data. To manage divergent search in this space, we rely on a meta-learning framework to integrate useful features from diverse tasks into a single model. The final layer of this model is used as an index into the "archive" of all past behaviors. We search for regions in the behavior space that the current archive cannot reach. As expected, divergent search is outperformed by models with a strong bias toward the evaluation tasks. But it is able to match and sometimes exceed the performance of models that have a weak bias toward the target task or none at all. This demonstrates that divergent search is a viable approach, even in high-dimensional behavior spaces.

4.2 Introduction

Unsupervised learning is attractive because it avoids the need for manual data labelling; but also because human-crafted objectives are often only proxies for desired behaviors. For image classification, self-supervision and clustering are some of the most popular approaches that do not depend on labels. A lesser known alternative is divergent discriminative feature accumulation (DDFA). It searches for features that partition the data in novel ways and accumulates these features in an archive [Szerlip et al., 2015]. The advantage of a divergent search is that it explores as many diverse partition behaviors as possible. Learning a wide range of behaviors helps prepare a system when the target task is not known *a priori*. In image classification, a single image can be classified in different ways based on its contents. An image of a dog at a park could be classified based on the breed of the dog, the dog's action/pose, the trees in the background, the season, the weather, or whether it is day or night time. Convergent algorithms

generally settle on one partition of the set and may fail to consider other possible arrangements. In many cases, image-space similarities dominate and more specific/subtle features are hard to discover without explicit supervision. We aim to use divergent search to continually seek novel partitions rather than converging to the cardinal dimensions of variation.

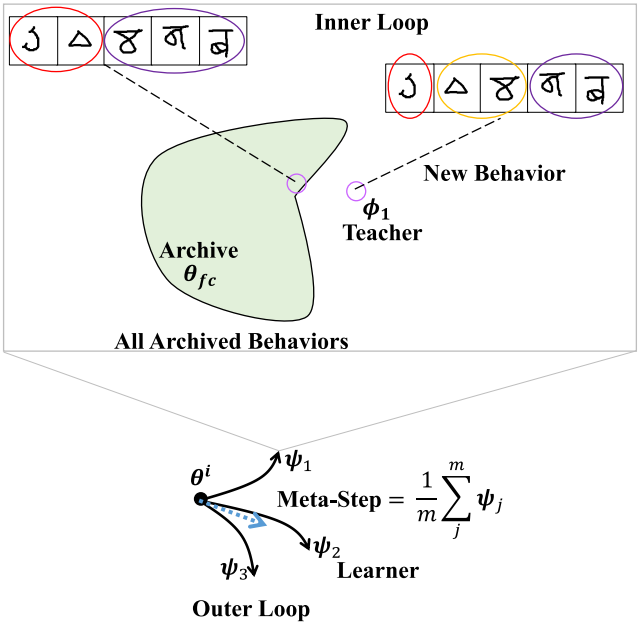


Figure 4.1: Overall framework with examples of Omniglot images. Each inner loop starts with the same model parameters, θ^i. Then a search for novel behaviors begins: a fully connected layer, θ_{fc}, acts as an index to all of the features stored in θ^i, which acts as an archive of past behaviors. A teacher model, ϕ_1, aims to find the parameterization of a behavior that is outside of the current capabilities of the archive θ^i. Then a learner ψ_1 is trained on that behavior. Finally the average of the gradients from each learner is used to update θ^i and the process repeats with an updated archive of behaviors.

Thus far, DDFA has only been demonstrated in very restricted settings (i.e. single layer features). One challenge of using divergent search with deep networks is the feature accumulation aspect. It is impractical to store copies of every network that produces a new partition behavior. It is even less feasible to search across a collection of networks for useful features to combine in the fine-tuning process. Recent work in meta-learning offers a simple way to learn from different tasks within a single network [Finn et al., 2017, Nichol et al., 2018]. These approaches aim to learn a set of features that is generally useful (or quickly adaptable) for a range of tasks, which may have never been presented during training. This lends itself well to divergent search.

Another challenge of using DDFA with deeper networks is the divergent search itself. Searching solely for novel partitions may suffice when the network's discriminative ability is inherently

limited (i.e. single-layer features), but deeper networks are able to learn parameter settings that perfectly match random labels [Zhang et al., 2017]. To prevent accumulation of arbitrary and low utility features, a measure of quality must be included in the divergent search. In the absence of ground truth labels, robust consistency between network outputs has become a popular surrogate for quality [Laine and Aila, 2017, Han et al., 2018]. We jointly search for novelty and quality by optimizing a teacher network to exhibit behaviors that

1. have robust consistency with a student model, but
2. cannot be learned by an archive of past behaviors (i.e. by reusing existing features)

Every novel behavior that is found in this search is integrated into the network. This increases its repertoire of discriminative behaviors and redefines novelty - pushing the search to new behaviors.

4.3 Related Work

This work aims to translate the motivation from DDFA into more modern meta-learning frameworks. Both approaches are outlined below.

Divergent Search

One of the seminal works in this area is novelty search [Lehman and Stanley, 2011a], which eschews performance-based objectives in favor of finding novel behaviors. Since the proposal of novelty search, divergent strategies have become a central component of many evolutionary methods, particularly in the field of quality diversity (QD). QD argues that the strength of evolutionary computation lies in discovering new behaviors rather than optimizing toward a single targeted objective [Pugh et al., 2016]. This paradigm has been used in many agent-based problem settings. For example, in robotics it has been used to learn a repertoire of behaviors spanning a robot's capabilities. This endows the robot with a range of pre-learned skills making

it better prepared for uncertain terrain or new, unseen tasks [Cully et al., 2015]. Note the close parallels with unsupervised learning and meta-learning.

Szerlip et al. proposed DDFA to harness divergent search for unsupervised image classification [Szerlip et al., 2015]. DDFA sets out to learn a repertoire of discriminative behaviors in hopes that some of these features will be useful for the target task. In their framework, each candidate feature is a single-layer feature (no hidden nodes) which outputs a scalar response for a given image. Behavior is measured as a feature's response to every image in the training set, forming a vector of length equal to the number of images. Novelty is then quantified as the distance between behavior vectors. In this way, new features are accumulated in an archive if their behavior vector is sufficiently different from existing features. Finally, a classifier can be trained on top of these features using labelled examples from a target task.

Single-layer features are not sufficient for most computer vision applications [Krizhevsky et al., 2012]. Scaling up to deeper networks is challenging because it is impractical to accumulate entire networks as opposed to individual features. The fine-tuning process also becomes unwieldy if the classifier must search for and combine features from an archive of networks. Furthermore, since deep networks have more discriminative capability than single-layer features, the space of possible behaviors could include virtually all possible partitions of the data. Running novelty search in this behavior space could lead to accumulation of features with very little semantic quality or utility.

Meta-Learning

Recent work in meta-learning could help to resolve some of DDFA's issues, specifically the integration of features for different tasks into a single network. Two very similar methods, model-agnostic meta-learning (MAML) [Finn et al., 2017] and Reptile (a play on words) [Nichol et al., 2018], aim to learn a good initialization point which can quickly adapt to new tasks. Both use an inner loop which learns from a single task, and an outer loop which aggregates information from multiple tasks. We choose Reptile as the basis for this work for its simplicity and computational efficiency. Reptile learns parameter settings for a task by taking several

ordinary gradient steps. This repeats for every task included in the outer loop, always restarting from the same initialization point. In the meta step, the initialization point is moved toward the mean of all task-tuned parameter settings.

Unsupervised Meta-Learning

While meta-learning approaches are typically used in supervised settings, they offer attractive properties for unsupervised learning. Of primary interest is their ability to learn and generalize from tasks that are *related*, but not *exactly* the same as the the target task. This gives us more lenience when generating tasks for training (in an unsupervised way). Hsu et al. have exploited this fact for the purpose of unsupervised meta-learning [Hsu et al., 2019]. Their approach is based on clustering to automatically generate tasks for unsupervised model-agnostic meta-learning (CACTUs-MAML). They use existing self-supervised and clustering based methods to generate *mock* tasks for training the meta-learning framework (in their case MAML). Note that the meta-learning model is typically small to prevent overfitting on few-shot tasks; but the self-supervised/clustering model can be much larger, creating a distillation effect [Hinton et al., 2015] from a more sophisticated model. The first step is to train a self-supervised/clustering method. Then k-means clustering is used to partition the *embeddings* of the entire training dataset, in the latent space of the self-supervised/clustering network. Using these clusters as labels, regular tasks can be created for training the meta-learning framework. Hsu et al. show that i) different self-supervised/clustering methods work better for different datasets and ii) the meta-learning performance is directly related to the quality of the mock tasks, i.e. the self-supervised/clustering method [Hsu et al., 2019]. As such, this process requires three distinct phases (self-supervision/clustering, k-means labelling, and meta-learning) with the potential need for trial-and-error in the self-supervised/clustering step. This process can be quite time-consuming and resource intensive.

To circumvent the arduous training of CACTUs-MAML, Khodadadeh et al. propose unsupervised meta-learning for few-shot image classification (UMTRA) [Khodadadeh et al., 2019]. Their method uses domain knowledge as a substitute for explicit labels to construct tasks that

are similar to the supervised tasks, but without the need for expensive labelling. First UMTRA exploits knowledge of the data distribution. Given the distribution of most few-shot datasets – these include many classes and only a small number of examples within each class – the samples in a small batch of images are likely to come from unique classes. UMTRA begins by sampling a small number of images and assigns unique labels to each image under the assumption that they are from unique classes. Their calculations indicate that a sample of five images from Omniglot has a 99.2% chance of containing five unique classes (and a chance of 85.2% for Mini-ImageNet) [Khodadadeh et al., 2019]. Data distributions of both datasets are shown in Table 4.1. The second source of domain knowledge is knowledge of identity preserving transformations. Khodadadeh et al. use data augmentation to artificially expand this small sample of images [Khodadadeh et al., 2019]. The best results are achieved when AutoAugment [Cubuk et al., 2019] is used, which optimizes an augmentation policy to increase accuracy on a validation set. This combination of i) statistically-favorable sampling and ii) artificial upsampling, aims to create tasks which are as similar as possible to supervised tasks.

State-of-the-art approaches use self-supervision, clustering, or domain knowledge to construct tasks that closely approximate supervised tasks. The goal of training on these tasks is to give a bias toward the type of behaviors that the model will be evaluated on. Training and testing on the same behavior is a cornerstone of machine learning. But recent works in evolutionary computation have asked the question of whether undirected search, which lacks any specific objective, can discover useful behaviors on its own. Many of these investigations co-evolve tasks and solutions together in an open-ended fashion. For example generating mazes and their solvers [Brant and Stanley, 2017] or courses with new terrains and agents that traverse them [Wang et al., 2019]. Similarly, we aim to learn teacher and student models that jointly explore image classification behaviors. This behavior space is very high-dimensional and lacks the interpretability enjoyed by many reinforcement learning problems (e.g. agent sensors/states which correspond to different actions, or environment parameterizations that correspond to different obstacles). Using divergent search in this type of behavior space provides insight into its suitability for such problems and the remaining unmet needs.

4.4 Methods

In this work, unsupervised learning is formulated as a divergent search for novel partition behaviors. A meta-learning approach inspired by Reptile [Nichol et al., 2018] forms the basis of our method. An inner loop is used to derive a single task, while an outer loop is used to aggregate learning from multiple tasks. The overall framework is depicted in Figure 4.1.

4.4.1 Inner-loop

Each iteration of the inner loop is an attempt to find a novel partition behavior. Three neural networks are used for this process. The **archive** is a fixed model which is parameterized by θ^i. At the start of each inner loop, all three networks are initialized to θ^i. Note that all models follow the same architecture, which is described later in Section 4.4.3. The final fully connected layer of the archive, θ_{fc}, is mutable. This means that the contribution of different higher-level features to a given logit can be changed. In this way, the fully connected layer acts as an index to all previous behaviors. This also includes exaptations of these archived features. In other words, features used for past behaviors can be repurposed for direct use in new behaviors. This allows the archive to span an even larger region of behavior space which includes all permutations of previous partition behaviors.

The **teacher** model aims to find regions of the behavior space outside of the subspace already spanned by the archive (green region in Figure 4.1). It is parameterized by ϕ. Unfortunately it is difficult to analytically determine the 'span' of the archive in behavior space. It is also prohibitively expensive to sample every possible index, θ_{fc}. As such, the archive index and teacher model are jointly optimized. The archive index is optimized to match the behavior of the teacher. Let $p_\phi(x)$ denote the teacher's prediction for image x where

$$p_\phi(x) = \arg\max f(x; \phi), \tag{4.1}$$

and let $f_\theta(x)$ denote the archive outputs:

$$f_\theta(x) = f(x; \theta^i, \theta_{fc}). \tag{4.2}$$

The archive index is then optimized to reduce the categorical cross-entropy between its outputs and the teacher's predictions:

$$\min_{\theta_{\text{fc}}} \; - \sum_{c=1}^{N=C} p_{\phi}\left(x\right) \log f_{\theta}\left(x\right) \tag{4.3}$$

Meanwhile, the teacher is optimized according to

$$\min_{\phi} \; -JS(f_{\theta}(x)||f_{\phi}(x)), \tag{4.4}$$

which aims to maximize the Jensen-Shannon (JS) divergence between the outputs of the archive and the teacher. The teacher and archive index are updated iteratively for a limited number of steps. In theory, a longer optimization process may find a better teacher behavior. To reduce computation time we restrict the number of iterations to 20 for all experiments. This is roughly double the number of steps that a model is typically given in the Reptile framework to train on a labelled classification task. As such, this limit is sufficient for learning new behaviors.

Another consideration is the optimizer, which in this case is Adam [Kingma and Ba, 2015]. Typically the momentum is disabled for the optimizer used in the Reptile framework because it leads to poorer performance [Nichol et al., 2018]. However, in our case we find it helpful to restore momentum for the teacher optimizer to avoid oscillations between different behaviors *within* the archive's reach.

Also note that the teacher is given a head-start by starting at a random mutation of the archive parameters through

$$\phi^{i} = \theta^{i} + s\eta. \tag{4.5}$$

Here η is the parameter space noise, sampled from a normal distribution $\mathcal{N}(0, \sigma)$. The standard deviation, σ, is set low and increased until the teacher behavior differs from the initial archive behavior by at least one prediction within a batch of images. Each parameter has a noise scaling factor s, which scales the noise to a reasonable range. This is estimated by 'safe mutations' which considers the magnitude of the gradient of all outputs with respect to a given weight [Lehman et al., 2018].

At the end of this process, a **learner** model is trained according to

$$\min_{\psi} \; -\sum_{c=1}^{N=C} p_{\phi}(x) \log f_{\psi}(x), \tag{4.6}$$

which attempts to match the teacher behavior. The learner model starts with the same parameters as the archive, $\psi^i = \theta^i$. The optimization of the learner is exactly the same as the standard inner loop in the Reptile framework [Nichol et al., 2018], but using the teacher predictions in place of ground truth labels.

4.4.2 Outer-loop

The standard outer loop for Reptile [Nichol et al., 2018] simply takes the final parameter values of the learner from each inner loop iteration and calculates the mean (Eqn 4.7). Then a meta-step is taken toward this mean, depicted by the blue arrow in Figure 4.1.

$$\overline{\psi} = \frac{1}{n} \sum_{i=1}^{n} \psi_i. \tag{4.7}$$

Using ground truth labels, it is natural to weight each task equally. In an unsupervised setting, however, some tasks may be higher quality than others. In fact, a critical step in QD is measuring the novelty and quality of a behavior to determine whether it should be kept or discarded. We assess novelty by measuring how accurately the archive is able to match the teacher behavior. Meanwhile quality is judged based on the robustness of the teacher behavior. To estimate robustness, we measure how accurately the teacher behavior can be matched when a model is trained on noisy examples of the teacher behavior. During the joint optimization of the archive index and teacher (described in Section 4.4.1), we include another copy of the archive that is trained on noisy teacher predictions (Eqn. 4.8). This is similar to Eqn. 4.3, except that i) optimization is no longer restricted to the last layer, θ_{fc}, and ii) the teacher predictions are affected by Bernoulli noise in the inputs to the fully connected layer ϕ_{fc}, i.e. dropout [Srivastava et al., 2014]. Using dropout in the final layer of the teacher will alter predictions that are less robust, while preserving predictions that are more robust.

$$\min_{\theta} \; -\sum_{c=1}^{N=C} p_{\phi}\left(x, n_{fc}\right) \log f_{\theta}\left(x\right) \;, \quad \text{where } n_{fc} \sim \text{Bernoulli}(0.5) \tag{4.8}$$

Accuracy for each copy of the archive is calculated according to Eqn. 4.9. The value of a behavior is then calculated as the difference in accuracy between the two archive copies which are optimized according to Eqn. 4.3 and Eqn. 4.8, which represent scores in novelty and robustness respectively. In other words, $V_{\phi} = A_{\text{robustness}} - A_{\text{novelty}}$. Note that a lower A_{novelty} corresponds to a more novel behavior because the archive index was not able to find a past behavior which accurately fit the teacher behavior.

$$A_{\theta} = \frac{1}{m} \sum_{j=1}^{m} p_{\theta}\left(x\right) == p_{\phi}\left(x\right). \tag{4.9}$$

If the behavior value is negative, the behavior is discarded and a new inner loop starts. The final set of positive values is normalized to have a unit sum. These values are then used to calculate a weighted average of the learners:

$$\overline{\psi} = \sum_{i=1}^{n} v_i \psi_i \;, \text{ where} \sum_{i=1}^{n} v_i = 1, v_i > 0. \tag{4.10}$$

This *weighted* meta-step has parallels with evolution strategies (ES) [Salimans et al., 2017]. In ES, samples in parameter space are evaluated using a potentially non-differentiable loss function. Then the update is found by averaging all parameter sets, weighted by their loss. In our case, the non-differentiable loss function aims to measure quality and diversity of behaviors, and the loss evaluation is a complete inner loop.

4.4.3 Model Architecture

The models used in our experiments follow the same architecture and hyperparameter settings as the original Reptile framework [Nichol et al., 2018]. A neural network is used with four convolutional layers, followed by a fully connected layer (Figure 4.2). Each convolution is

followed by batch normalization and a ReLU activation. In the case of Mini-ImageNet, there is also a max-pooling operation before the ReLU. The convolutional layers have 64 feature channels for the Omniglot dataset and 32 for Mini-ImageNet.

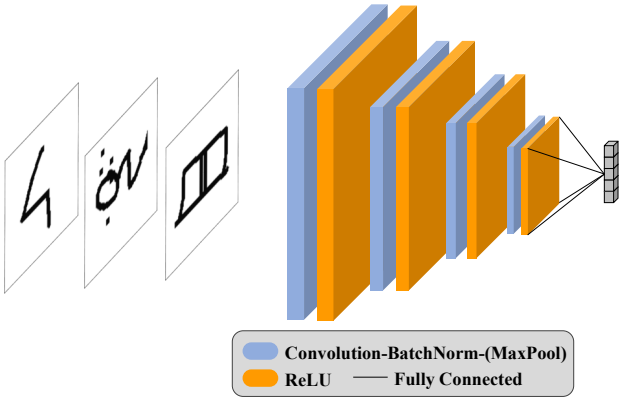


Figure 4.2: Network architecture with Omniglot example inputs.

4.5 Experimental Studies

4.5.1 Datasets

Two of the most commonly used datasets for few-shot learning are Omniglot [Lake et al., 2015] and Mini-ImageNet [Vinyals et al., 2016, Russakovsky et al., 2015]. Each of these datasets consists of many classes and relatively few examples (see Table 4.1). Omniglot images are hand-written characters from many different alphabet systems, captured in grayscale with dimensions $28x28$ pixels. Mini-ImageNet consists of natural images of different subjects (e.g. golden retriever, poncho, school bus) captured in color with dimensions $84x84$ pixels.

Table 4.1: Dataset specifications.

Dataset	Omniglot	Mini-ImageNet
Pre-training Classes	1200	64
Evaluation Classes	423	36
Images per Class	20	600

4.5.2 Image Sampling

An important consideration is how images are sampled for each inner loop. As mentioned in Section 4.3, UMTRA exploits knowledge of the data distribution to sample small batches which are likely to contain images from different classes. Figure 4.3 illustrates that UMTRA's statistically-favorable sampling, combined with augmentation, simulates genuine class-based sampling. Instead of relying on statistically-favorable sampling, we wish to simulate a case where sample size can no longer be exploited. The probability that each sample in a batch comes from a different class is

$$P = \frac{c! \cdot m^N \cdot (c \cdot m - N)!}{(c - N)! \cdot (c \cdot m)!}, \tag{4.11}$$

where c is the number of classes, m is the number of images in each class, and N is the sample size [Khodadadeh et al., 2019].

By using a sample size of 20 images for Mini-ImageNet and 90 images for Omniglot, we can reduce the chance that all images are from unique classes to about 3%. We use this setting, depicted in the last row of Figure 4.3, for the proposed method. This distribution will likely contain multiple images from the same class, but it will also likely contain more classes than available logits. This means that pre-training tasks will almost certainly violate class-boundaries. These conditions are not favorable for inducing a bias toward class-oriented evaluation tasks. However, it simulates the most general case of unsupervised learning, where the number of classes and the distribution among classes is not known *a priori*.

4.5.3 Evaluation

A few-shot learning task is used to evaluate whether the algorithm has learned useful features. In this setting, N classes are selected and $K + 1$ images are sampled from each class. The model is given the first K images (from each class) along with their ground truth labels. After the model is fine-tuned using these labelled examples, the model must predict the correct class for the final (+1) image in each class. Note that 'fine-tuning' will be used to refer to the

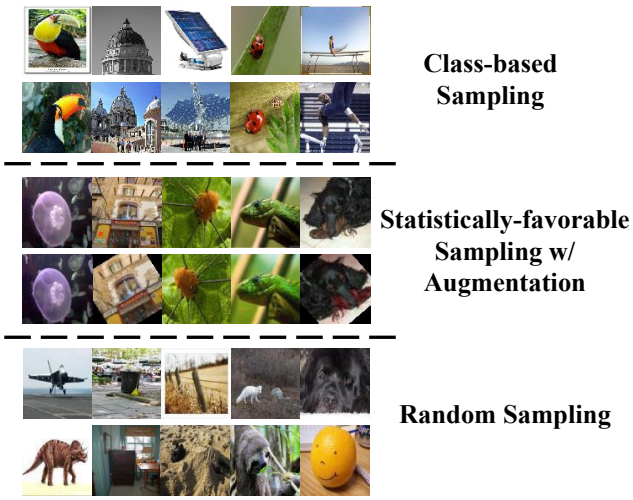


Figure 4.3: Examples from Mini-ImageNet based on different sampling schemes. Each row represents a batch.

learning that occurs during the evaluation stage (using the K labelled examples); meanwhile 'pre-training' will be used to refer to all learning that occurs before the evaluation stage. The reported accuracy metric is the average across 10,000 tasks, where each task uses a random set of N classes. Note that the pre-training classes and test classes are disjoint sets.

4.5.4 Benchmark Methods

Most existing approaches, whether supervised or unsupervised, attempt to give the model a bias toward tasks that are similar to the evaluation tasks. In contrast, the proposed method is not biased toward any particular type of classification, but instead tries to learn as many discriminative behaviors as it can. A major question that arises is: *whether the features learned by an undirected, divergent search are useful for downstream tasks designed by humans.* In particular, how does it compare with methods that induce a strong/weak bias toward the evaluation tasks or indeed away from the evaluation tasks. We use four main benchmarks to represent biases of varying strength: i) fully supervised, ii) UMTRA, iii) random initialization, and iv) random labels.

Fully Supervised

The fully supervised setting (Reptile [Nichol et al., 2018]) represents the most explicit way to create an inductive bias toward tasks that are similar to the evaluation tasks. Supervised pre-training uses labelled images to learn tasks that are essentially equivalent to the evaluation tasks, but using different classes.

Unsupervised meta-learning for few-shot image classification.

UMTRA uses domain knowledge as a substitute for explicit labels to construct tasks that are similar to the evaluation tasks. It combines a) statistically-favorable sampling and b) artificial upsampling through data augmentation [Khodadadeh et al., 2019]. We re-implement UMTRA within the Reptile [Nichol et al., 2018] framework for a more direct comparison. The original UMTRA uses AutoAugment [Cubuk et al., 2019] to create new realistic samples. AutoAugment has recently surpassed state-of-the-art scores on formidable challenges such as ImageNet [Russakovsky et al., 2015], purely through data augmentation. In our case, we restrict augmentations to a more modest set of standard transformations, including random combinations of:

- Horizontal flipping
- Hue, saturation, brightness and contrast adjustment by a random factor within ranges [-0.08,0.08],[0.6,1.6],[-0.05,0.05], and [0.7, 1.3] respectively
- Random rotation by an angle within $[-\frac{\pi}{4}, \frac{\pi}{4}]$
- Random cropping ranging from 1% to 20%

While the original UMTRA uses strong augmentation to effectively 'generate' additional examples of each class, our low-augmentation setting is more akin to teaching invariance to simple transformations. We use this baseline to represent a weaker bias toward evaluation tasks and include the original UMTRA results for reference.

Random Initialization

Random initialization represents an unbiased baseline, i.e. the performance of a model without pre-training. This method learns only from the fine-tuning during evaluation.

Fixed Random Labels

The fixed random labels approach represents a bias *away* from the evaluation tasks. Assigning fixed random labels to the images creates new tasks that are very unlikely to overlap with the type of tasks used in evaluation. In these randomly labelled tasks, images with the same label are likely to come from different classes and the same class may appear under more than one label. As such, class-specific features are no longer reliable signals for discrimination. Instead, the model may rely on extraneous information in the images (e.g. noise or background objects). Note that, in some cases, pre-training on random labels (i.e. 'memorizing') can actually lead to an increase in accuracy for a supervised task as compared with starting from a random initialization [Pondenkandath et al., 2018].

To summarize, (i) fully supervised and (ii) low-augmentation UMTRA represent strong and weak biases toward the evaluation tasks respectively; (iii) random initialization represents an unbiased starting point; and (iv) random labels acts as a bias away from the evaluation tasks. Comparing a divergent strategy to this spectrum of pre-training approaches allows us to assess how useful the discovered features are for real, human-designed tasks.

4.6 Results

Evaluation was performed using $N = 5$ classes and $K = 5$ fine-tuning examples ($k = 1$ was also tested). The results are summarized in Table 4.2 and Table 4.3. The primary benchmark methods are shown in bold and below them are additional results summarized from literature [Hsu et al., 2019, Khodadadeh et al., 2019]. Note that bidirectional generative adversarial network (BiGAN) [Donahue et al., 2017], adversarially constrained autoencoder in-

terpolation (ACAI) [Berthelot et al., 2019], and DeepCluster [Caron et al., 2018] are different self-supervised/clustering approaches. The representation learned from these methods can be used in several ways, e.g. a linear classifier trained on top of the representation or a k-nearest neighbors approach. Here, only the best and worst scores are included. For full details, refer to the original work by Hsu et al. [Hsu et al., 2019]. Note that despite the effort required to train these various methods, several of them produce scores below random initialization.

Table 4.2: Omniglot results for $N = 5, K = 5$ and $N = 5, K = 1$ evaluation tasks. Bottom results summarized from [Hsu et al., 2019, Khodadadeh et al., 2019].

Omniglot				
Algorithm (way-shot)	(5,1)		(5,5)	
	train	test	train	test
Fixed Random Labels	21.11	20.84	26.05	25.60
Random Initialization	34.11	33.64	66.28	64.29
Proposed	59.53	55.79	80.87	77.85
UMTRA-Reptile Low Aug	66.21	65.26	85.32	83.48
Fully Supervised	98.32	97.64	99.54	99.68
Random Initialization MAML	- -	52.50	- -	74.78
BiGAN worst scores	- -	40.54	- -	58.52
BiGAN best scores	- -	49.55	- -	68.72
BiGAN CACTUs-MAML	- -	58.18	- -	78.66
ACAI worst scores	- -	51.95	- -	71.09
ACAI best scores	- -	61.08	- -	81.82
ACAI CACTUs-MAML	- -	68.84	- -	87.78
UMTRA-MAML	- -	83.80	- -	95.43

For both datasets, the proposed divergent search strategy achieves similar performance to methods with a weak bias toward evaluation tasks (low-augmentation UMTRA). However, methods with a stronger bias (fully supervised, original UMTRA, and CACTUs-MAML methods), still perform significantly better. Meanwhile the fixed random labels approach actually degrades performance compared to an unbiased start from randomly initialized parameters.

A closer examination of the classification behaviors requires inspection of individual tasks. Figure 4.4 displays several example tasks using t-distributed stochastic neighbor embedding (t-SNE) plots [Maaten and Hinton, 2008, Policar et al., 2019]. In each case, the model is fine-tuned on $K = 5$ examples from $N = 5$ classes. Then 10 test images are sampled from each class and the logit outputs are visualized using t-SNE. Although t-SNE is not deterministic, the resulting plots provide a *potential* interpretation of the model's sorting criteria. Red and green borders

Table 4.3: Mini-ImageNet results for $N = 5, K = 5$ and $N = 5, K = 1$ evaluation tasks. Bottom results summarized from [Hsu et al., 2019, Khodadadeh et al., 2019].

Mini-ImageNet				
Algorithm (way-shot)	(5,1)		(5,5)	
	train	test	train	test
Fixed Random Labels	25.78	24.46	36.97	32.49
Random Initialization	26.02	24.59	39.93	36.52
Proposed	33.74	30.90	47.58	43.41
UMTRA-Reptile Low Aug	34.13	31.45	46.84	42.15
Fully Supervised	64.24	50.65	78.52	66.33
Random Initialization MAML	- -	27.59	- -	38.48
BiGAN worst scores	- -	22.91	- -	29.06
BiGAN best scores	- -	27.08	- -	33.91
BiGAN CACTUs-MAML	- -	36.24	- -	51.28
DeepCluster worst scores	- -	22.20	- -	23.50
DeepCluster best scores	- -	29.44	- -	42.25
DeepCluster CACTUs-MAML	- -	39.90	- -	53.97
UMTRA-MAML	- -	39.92	- -	50.73

Table 4.4: Ablation study on Mini-ImageNet with $N = 5, K = 5$ evaluation tasks.

Ablation Study		
Algorithm	Accuracy	
	train	test
Random Search	32.54	26.63
Random Initialization	39.93	36.52
Proposed (Diversity Only)	46.52	42.41
Proposed (Quality Diversity)	47.58	43.41

around each image indicate the classification outcome (incorrect/correct respectively). In some instances, classification outcome is poor, but the images are still embedded in meaningful ways.

We also perform an ablation study (Table 4.4). First, the quality aspect is removed by accepting every behavior found by the divergent search in the inner loop ('Divergent Only'). Then, the divergent search is replaced by random search. In random search, labels are sampled dynamically from a discrete uniform distribution across all possible classes. Note that this is different from the *fixed* random labels setting where an image is assigned to a fixed group for the entire pre-training process. Each component of the proposed method increases the evaluation accuracy, especially the divergent search used in the inner loop.

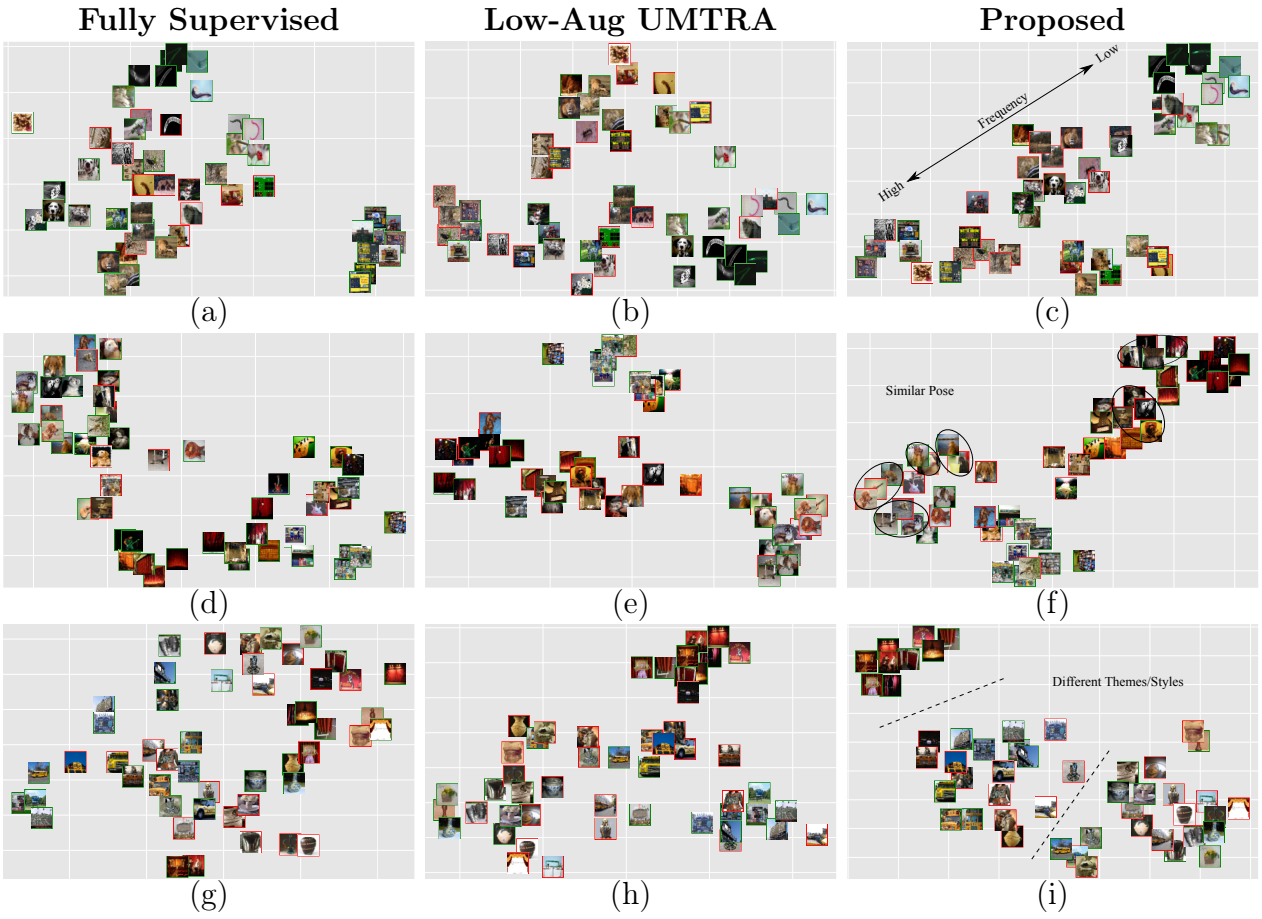


Figure 4.4: Example t-SNE plots using the logit outputs for 10 test images per class after fine-tuning on $N = 5, K = 5$ examples. Each row is a different task. Green/red borders indicate whether images were correctly/incorrectly classified. The fully supervised approach is able to group images by class characteristics that are invariant to transformations. The proposed divergent approach appears to rely on more generic features such as spatial frequency, pose, or image style. Best viewed digitally.

4.7 Discussion

Evaluation on class-based tasks indicates the usefulness of a model's learned features. Supervised tasks and tasks that are constructed to *approximate* supervised tasks lead to the highest performance. Divergent search is able to learn features which are more beneficial than harmful. However, as expected, its performance is still significantly lower than methods with a strong bias toward evaluation tasks. The image sampling protocol (Section 4.5.2) used in the proposed method produces batches of images that do not neatly fit into five classes. Nonetheless, the proposed algorithm tries to find a new, high quality behavior for each batch. In contrast, UMTRA relies on statistically-favorable sampling and CACTUs-MAML draws samples from distinct re-

gions in an embedding space, all in an effort to approximate supervised tasks. These results demonstrate that *despite* unfavorable conditions, an undirected search in a high-dimensional behavior space can still be fruitful. In comparison, many of the learned embeddings (BiGAN, ACAI, DeepCluster) perform worse than divergent search, sometimes even worse than random initialization. These learned embeddings typically find features that are relevant to the data; but without a bias toward evaluation tasks, their efforts can be detrimental.

The random fixed labels approach also leads to a drop in performance (Table 4.2 and Table 4.3). This suggests that the model is specializing in a behavior that is counterproductive. Hsu et al. found a similar reduction in performance when training on random labels [Hsu et al., 2019]. These random tasks often include a) multiple classes under the same label and b) the same class under multiple labels. As such, it is reasonable that the bias learned from this pre-training is harmful when performing class-based evaluation tasks. However, the tasks constructed in a divergent search are also likely to violate class-boundaries. In spite of this, the divergent search is able to learn features that are more beneficial than harmful. One key consideration is that optimization on *fixed* random labels will lead to specialized discriminative behaviors for *that* distribution of tasks. In contrast, a divergent search tries to explore as many novel discriminative behaviors as possible.

Ideally, a divergent search should accumulate specialists in *many* different behaviors. But in this setting the model size is restricted for fair comparison with benchmarks. It is also impractical to use niche-preserving methods (e.g. novelty search with local competition [Lehman and Stanley, 2011b]) when dealing with such a high-dimensional behavior space. As such the model is incapable of learning a specialized set of behaviors for every task it encounters as niche-preserving methods do. Instead, the best use of the model capacity may be a set of generic features that have the most overall utility. Figure 4.4 shows that the proposed method may have learned to distinguish images by their spatial frequency, pose, and style, all of which are fairly generic descriptors. This type of undirected learning may eventually lead to more robust, general-purpose features than those found by tuning a model precisely to human-crafted objectives. However, the wide gap in performance when compared to strongly biased methods suggests that its current state leaves much to be desired.

The fully supervised method achieves higher scores by grouping images based on characteristics associated with class identity. For example, in Figure 4.4 panel (a), the model groups lions and Dalmatians separately despite their high intra-class variance. Low-augmentation UMTRA, panel (b), is also able to make this distinction. Both have learned some amount of invariance to identity-preserving transformations thanks to the bias in their pre-training strategies.

Divergent search is also often compared with random search. In this case, the tasks for random search are constructed by dynamically sampling labels from a discrete uniform distribution over all possible classes. Sampling random tasks in this way effectively asks the model to perform diverse random behaviors. Uniform random search in behavior space has recently been proposed as an interpretation of the overall outcome of novelty search [Doncieux et al., 2019]. Given these intuitions, random search might be expected to perform just as well as a divergent search. However, Table 4.4 indicates that this formulation of random search actually has a deleterious effect on evaluation performance. The main difference between a random behavior task and a divergent task is that the divergent task is strongly connected to the model's current capabilities. In our case, the divergent task originates from a task that the model can already perform. Then it is optimized away from the space of behaviors that the model can achieve given its current features. This optimization process is stochastic, heuristic and imperfect. The resulting task may lie somewhere between easy (the model's current behavior) and hard (a behavior that is impossible with the model's current features). This may unintentionally regularize the difficulty of divergent tasks and form a sort of curriculum [Bengio et al., 2009]. As such, these divergent tasks may have more structure than the random search in high-dimensional behavior space, allowing for more constructive learning.

Our experiments explore the application of divergent search to image classification. Although the motivation is similar to many existing works on divergent search, there are several key differences. One difference is in the nature of the task. In maze solving tasks, which is a common problem setting for divergent and open ended evolution studies [Lehman and Stanley, 2011a, Brant and Stanley, 2017], the complexity of a maze is fairly simple to measure and provides a meaningful metric of progression. This is useful for creating a trivial starting point (e.g. empty maze) for behaviors to develop. It also inherently provides a direction for learning which can

gradually increase in complexity, e.g. sequentially adding more barriers. Neither of these are available in the image classification setting. A trivial behavior could be classifying all images as the same class, but this is actually a degenerate behavior as compared to an initialization from random [Glorot and Bengio, 2010] parameters. Furthermore, task complexity is not easily measurable and more complex tasks do not lead to better or more useful behaviors. Another difference is that the behavior space is very high-dimensional compared to the environments used in many reinforcement learning problems. In agent-based frameworks, the behaviors can often be conflated onto a low-dimensional space where nearly all regions are meaningful [Lehman and Stanley, 2011b]. Exploring this behavior space is very likely to be fruitful. In the case of image classification, the behavior space is vast and many regions of the space do not correspond to anything meaningful. Although this is a challenging setting, the behavior space found in nature is even larger. Bringing divergent search into these larger playgrounds can provide insight into the needs that are currently unmet.

4.8 Summary

Divergent search is a major component of QD methods which aim to learn a repertoire of behaviors in the absence of specific behavioral objectives. The reasoning behind this approach is that a system equipped with a whole host of skills will be able to handle novel tasks better than a system which is tuned for one specific task. Meta-learning, i.e. learning how to best learn a new task, has a very similar end goal. However it is typically approached from the angle of generalizing from tasks that are similar to the target task. In this work we investigate the use of divergent search in a few-shot image classification setting. Comparing divergent search to meta-learning approaches, which are biased toward the target task, reveals a significant gap in performance. But compared to weaker biases or methods which are not specifically biased, divergent search is often comparable or better. These results demonstrate that divergent search is a viable approach, even in a high-dimensional behavior space. There is also considerable room for improvement, particularly in the measurement of behavior quality. In future work, better quality metrics could help condense the behavior space to dimensions with more utility. A

more modular architecture could also preserve archived behaviors in a better way, facilitating exaptation.

Another takeaway from these experiments is that simple biases toward the target task can have a huge impact on performance. General, unbiased feature learning is certainly a meaningful endeavor. But practical solutions must often accept some drawbacks to achieve better performance within a narrower scope. The following chapters explore more targeted solutions that are specifically tailored to medical outliers.

Chapter 5

Detecting Outliers with Foreign Patch Interpolation

Publications Associated with this Chapter

[1] Tan, J., Hou, B., Batten, J., Qiu, H., Kainz, B., et al. (2022). Detecting outliers with foreign patch interpolation. *Machine Learning for Biomedical Imaging*, 1(April 2022 issue):1–10

This chapter presents a novel method for detecting outliers called foreign patch interpolation [Tan et al., 2020]. This method achieved first place in the 2020 medical out-of-distribution Analysis (MOOD) challenge in both sample and pixel level categories [Zimmerer et al., 2020a]. The MOOD challenge is run under the medical image computing and computer assisted intervention (MICCAI) conference. It has also been published in the journal of machine learning for biomedical imaging (MELBA).

The divergent search method proposed in the previous chapter is theoretically interesting, but lacks the performance required for practical applications. Self-supervision is one of the best strategies for learning useful representations. The types of features that are learned depends completely on the self-supervised task. As such, the goal is to craft a task that pertains to the most relevant features. The proposed method takes a normal image and introduces a "foreign" irregularity from another normal image. The network is then trained to estimate the degree of deviation from normal. This self-supervised task encourages the network to learn normal

features and to detect subtle deviations.

5.1 Overview

In medical imaging, outliers can contain hypo/hyper-intensities, minor deformations, or completely altered anatomy. To detect these irregularities it is helpful to learn the features present in both normal and abnormal images. However this is difficult because of the wide range of possible abnormalities and also the number of ways that normal anatomy can vary naturally. As such, we leverage the natural variations in normal anatomy to create a range of synthetic abnormalities. Specifically, the same patch region is extracted from two independent samples and replaced with an interpolation between both patches. The interpolation factor, patch size, and patch location are randomly sampled from uniform distributions. A wide residual encoder decoder is trained to give a pixel-wise prediction of the patch and its interpolation factor. This encourages the network to learn what features to expect normally and to identify where foreign patterns have been introduced. The estimate of the interpolation factor lends itself nicely to the derivation of an outlier score. Meanwhile the pixel-wise output allows for pixel- and subject-level predictions using the same model.

5.2 Introduction

Outliers in medical data can range from obvious lesions to subtle artifacts. This wide range can make it difficult for a single detection system to identify all irregularities. Moreover, examples of outliers are often not available before testing takes place. This makes it difficult to use conventional classification methods that rely on training data to learn how to recognize test images that come from the same distribution. Without knowing what to look for, this task can be challenging even for human radiologists. For example, when focused on a lung nodule detection task, 83% of radiologists failed to notice a gorilla superimposed on the image ([Drew et al., 2013]). This indicates that human attention can cause even experts to be blind to

unexpected stimuli. It is infeasible to have radiologists repeatedly scan for every conceivable irregularity. As such, there may be an opportunity for automated systems to support detection, especially if these tools can offer a complementary view of the data.

Recent works have used neural networks to create high performance image recognition systems. These systems typically learn to recognize different image classes based on features that distinguish them from each other. However, outlier classes are not available during training, so it is not known *a priori* which features will be most relevant.

To circumvent this issue, reconstruction-based methods ([Baur et al., 2020, Zimmerer et al., 2019a, Alex et al., 2017, Schlegl et al., 2017]) aim to learn a complete model of the normal data. Abnormalities are then found by comparing the original image to its reconstruction. A key limitation of this approach is that it directly compares pixel intensities under the assumption that intensity differences will be proportional to abnormality.

Self-supervised methods offer an alternative approach to feature learning. These methods employ data augmentation techniques to create their own labelled samples from unlabelled data. Methods such as geometric transformations ([Golan and El-Yaniv, 2018]) have been successfully applied to outlier detection and outperform reconstruction-based methods on datasets with high variation such as CIFAR-10 ([Krizhevsky, 2009]). These methods can tolerate higher variation in normal data because they learn to identify salient structures in the normal class rather than relying on precise reconstruction of every pixel. However, for most cases in medical imaging, all of the major anatomical structures are present, even in abnormal cases. To detect medically relevant outliers, we aim to develop a method that can tolerate variations of normal anatomy while also being sensitive to fine-grained deviations from normal.

We propose a self-supervised task to train a model to learn where and to what degree a foreign pattern has been introduced. The goal is to encourage the model to learn what features to expect normally, given the context, and to be sensitive to subtle irregularities.

We evaluate this approach on an internal evaluation set with synthetic abnormalities and submitted the technique to the 2020 MICCAI medical out-of-distribution (MOOD) analysis chal-

lenge ([Zimmerer et al., 2020b]) where it ranked first in both sample and pixel level tasks. We also evaluate our method's ability to detect real medical anomalies using the DeepLesion dataset ([Yan et al., 2018a]).

5.3 Related Work

Out-of-distribution (OOD) detection is a broad topic discussed by many communities ([Pimentel et al., 2014, Pang et al., 2020]). Depending on the context, OOD samples may contain minute defects or completely unrelated content. It is often hard to formally define what constitutes an OOD sample, especially without any reference examples. This makes the task inherently heuristic and each approach must accept some assumptions which will impact its ability to detect different types of outliers. One strategy is to choose assumptions that will generalize as broadly as possible and be sensitive to the types of outliers that are of most interest. Most existing methods detect outliers based on reconstruction error, embedding space distances, or more recently, performance on self-supervised tasks.

Before discussing unsupervised methods, it is important to note that there are many supervised and semi-supervised methods for detecting abnormalities. Supervised methods have achieved expert-level performance in detecting breast cancer ([Wu et al., 2019]), retinal disease ([De Fauw et al., 2018]), pneumonia and other chest abnormalities ([Tang et al., 2020]). Some of these methods also delineate the boundaries of abnormalities, *e.g.*, brain tumor segmentation ([Menze et al., 2014]). Typically, these supervised methods learn from labelled examples of the target class and are not designed to generalize to other types of abnormalities. Alternatively, there are outlier detection methods that use labelled examples from a subset of anomalies with the goal of detecting broader classes of outliers. One example is outlier exposure ([Hendrycks et al., 2019]), which trains a multi-class classifier on several classes of normal data and tunes the network to make less confident predictions on a set of OOD training samples that do not belong to any of the normal classes. This tuning can help the model to make less confident predictions on OOD samples, even if they come from a different distribution than the OOD training samples.

However, for medical anomalies, which can be very subtle, it is not always possible to obtain a relevant OOD training dataset. Since our proposed method uses only normal samples, we focus on comparing to similar unsupervised methods described below.

Reconstruction-based methods attempt to reproduce images using a model of the normal data. This model may be characterised by the bottleneck of an autoencoder ([Atlason et al., 2019]) or variational autoencoder (VAE) ([Zimmerer et al., 2019a]) or by the latent space of a generative adversarial network (GAN) ([Schlegl et al., 2017]). Reconstruction-based methods are especially common in medical imaging applications. They allow for pixel-level localization and offer some level of interpretability through the reconstructed images. Baur et al. provide a comparative study with many variants of reconstruction-based methods using brain MRI data ([Baur et al., 2021]). Autoencoders are versatile and easy to implement across a wide range of datasets and configurations. For example, different variations have been applied to chest X-ray ([Mao et al., 2020]), mammography ([Wei et al., 2018]), and brain CT ([Pawlowski et al., 2018]) data. Unconstrained, autoencoders run the risk of reconstructing anomalies along with normal anatomy. As such, many methods use some form of regularization on the latent representation. For instance, [Zimmerer et al., 2019a] use a VAE, which maps samples to distributions over the latent space and minimizes the Kullback-Leibler divergence between the approximate posterior and a prior. Alternatively, a discriminator can be used to match the distribution of latent codes to a prior; this type of adversarial autoencoder has also been used in outlier detection ([Chen and Konukoglu, 2018]). Another option is to eliminate the bottleneck entirely by using a GAN to learn the distribution of normal data. To reconstruct a query image, a latent code can be optimized to find the best match within the learned distribution ([Schlegl et al., 2017]) or an encoder can be learned to map images directly into latent codes in a single step ([Schlegl et al., 2019]). There are also restorative methods that replace low likelihood regions in the image with samples from a learned prior ([You et al., 2019, Marimont and Tarroni, 2021]). As such, there are multiple strategies for reconstructing the normal components of the input image. Any errors in the reconstruction are then used to highlight anomalies. However, this means that the abnormality score is proportional to intensity differences in the input space. This neglects some of the key advantages of deep learning. Primarily, it fails to make use of learned mappings

that bring raw inputs into representations where semantic differences can be distinguished more easily ([LeCun et al., 2015]).

All of the above methods use whole images, but the reconstruction task can also be simplified to focus on patterns at a smaller scale. Patch-level reconstruction can be effective for detecting pathological textures in mammograms ([Wei et al., 2018]). Decomposing an image into smaller patches can also make it easier to train models, such as GAN's, without down-sampling or losing high-resolution texture information ([Alex et al., 2017]). Even if a model is trained at the patch level, anomaly scores can be recovered at the pixel level by using overlapping patches during inference ([Alaverdyan et al., 2020]). Some of these methods are trained using autoencoder or GAN losses, but exploit components other than the reconstruction error to compute anomaly scores. These can include the discriminator of a GAN ([Alex et al., 2017]) or the latent representation of an autoencoder ([Alaverdyan et al., 2020]). Using the embeddings of an encoder has the potential to facilitate semantic distinctions. However, if the encoder is not trained with an appropriate loss, then the representation may not distinguish relevant samples. For example, a discriminator is trained to separate real and generated samples. This does not necessarily make the representation suitable for separating real healthy samples from real pathological samples.

Other approaches train encoders using losses that are specifically designed for outlier detection. One example of this learns to map training samples to a compact sphere ([Ruff et al., 2018]). However, without any examples of outliers in the training data, this latent space may accentuate the wrong features, *i.e.*, variations within the normal data that are class invariant. Some embedding approaches introduce a disjoint set of outlier examples ([Bozorgtabar et al., 2020]) to overcome this issue. However in this work we focus on methods using only normal data.

Self-supervised methods have recently become a popular approach for unsupervised feature learning, especially variants of contrastive predictive coding (CPC) ([Oord et al., 2018, Hénaff et al., 2019]). Self-supervised methods have also been used for outlier detection ([Golan and El-Yaniv, 2018]), in some cases also combined with CPC ([Tack et al., 2020]). The main principle underlying many of these methods is to transform the images (*e.g.*, rotation) and

train a network to identify the transformation. This will sensitize the network to any features that change consistently with the transformation. For example, the brainstem (in a coronal view) may provide a reliable signal for predicting image rotation. However, if the brainstem structure is missing or occluded, the prediction accuracy may go down, indicating a potential outlier. This approach works well for recognizing key characteristics present in normal data. However, in medical images many pathological outliers may still conform to the same global structure as normal data.

Data augmentation and image synthesis play important roles in several outlier detection methods including our proposed method. In natural image datasets, data augmentation has been used to apply affine transformations, blur or sharpen images, or alter the color, brightness, and contrast of images. Methods such as AutoAugment and RandAugment find the most suitable combination of transformations and achieve state-of-the-art performance on supervised tasks through data augmentation alone ([Cubuk et al., 2019, Cubuk et al., 2020]). For medical imaging applications, elastic deformations and image synthesis can help generate more relevant or realistic augmentations ([Nalepa et al., 2019]). Some methods even model artifacts from the imaging modality used for data acquisition, *e.g.*, the bias field in MRI ([Chen et al., 2020a]). The data augmentation method that is most closely related to ours is Mixup ([Zhang et al., 2018]), which has previously been applied to improve brain tumor segmentation ([Eaton-Rosen et al., 2018]). Mixup creates convex combinations of samples and their respective labels. This helps regularize the network to behave linearly in-between classes. It also improves generalization and robustness to adversarial examples. Similarly, CutMix ([Yun et al., 2019]) works by copying a patch from one image and placing it into another image. The labels from both of these images are then mixed (as a convex combination) using a mixing factor equal to the patch area divided by the total image area.

Both Mixup and CutMix use convex combinations of ground truth labels. However, when there is only one class, which is the case in outlier detection, these convex combinations become meaningless. Self-supervised methods solve this problem by creating new classes through augmentations, *e.g.*, geometric transformations. However, these methods detect outliers through a proxy task, *i.e.*, classifying transformations, instead of directly identifying deviations from normal.

This can make it harder to recognize more fine-grained, localized irregularities. Classification-based proxy tasks also lack a direct means of locating abnormalities in the image. In this work, we show that these elements can be combined in a novel way, using convex combinations to create a new class that represents abnormality. This allows us to train directly on the task of estimating deviation from normal. Meanwhile, our patch-level augmentation setup naturally lends itself to pixel-level localization.

We provide the full details of our proposed method in the following section. Compared to existing methods, our self-supervised task is designed specifically to improve sensitivity to subtle irregularities. We target these cases because 1) they may be more medically relevant and 2) detecting them may be more useful to radiologists since fine-grained outliers typically require more intense scrutiny, time, and energy to detect.

5.4 Method

Most self-supervised methods train a network on a proxy task (*e.g.*, identifying geometric transformations ([Golan and El-Yaniv, 2018])) and subsequently measure abnormality as *failure* to perform this task. Many of these tasks are helpful for detecting the presence (or absence) of prominent structures that appear in the normal class. But medical images often contain more fine-grained outliers, where most major structures are still intact. As such, we propose a patch-level self-supervision task.

To create a variety of subtle outliers we extract the same patch from two independent subjects and replace the patch with an interpolation between both patches. The operation is shown in Eqn. 5.1 where A and B are independent samples, i refers to individual pixels in a patch h, and α is the interpolation factor. Note that A, B, and A' are full sized images. Pixels outside of the patch remain unchanged and whole images are used as inputs. The patch size, h_s, patch center coordinates, h_c, and the interpolation factor are all randomly sampled from uniform distributions (Eqn. 5.2-5.4). The pixel coordinates of the patch define the region that will be extracted from both samples, A and B. For volumetric data, each slice is paired with the

corresponding slice from a second subject, based on slice indices. For 2D data or data without a uniform number of slices, images are paired randomly. In both cases, we do not perform any registration preprocessing steps on the data. Instead, we exploit the natural variations and misalignment to create diverse training examples. Patches are square unless truncated by image boundaries or in pixels where A and B have the same value. Patch width ranges between 10% and 40% of the image width, d.

$$A'_i = (1 - \alpha)A_i + \alpha B_i \ , \ \forall \ i \in h \tag{5.1}$$

$$h_s \sim U(0.1 \cdot d, 0.4 \cdot d) \tag{5.2}$$

$$h_c \sim U_2(0.1 \cdot d, 0.9 \cdot d) \tag{5.3}$$

$$\begin{aligned} &\alpha \sim U(0, 1) \text{ for continuous } \alpha \text{ or} \\ &\alpha \in \{0, 0.25, 0.50, 0.75, 1\} \text{ for discrete } \alpha \end{aligned} \tag{5.4}$$

Although A and B are both normal on their own, the differences between them will cause the interpolation, A', to have artificial defects. We train a network to estimate where, and to what degree, a foreign pattern has been introduced. Given A' as input, the corresponding label includes the patch, h, and the interpolation factor, α, in the form of pixel-level values (Eqn. 5.5). The loss is thus a pixel-wise regression if α is continuous, or a pixel-wise classification if α is discrete. In both cases a standard cross-entropy loss is used (Eqn. 5.6-5.7, where f represents the model). For continuous α, cross-entropy operates on labels that are not one-hot; this is similar to applications such as label smoothing ([Szegedy et al., 2016]), network distillation with soft targets ([Hinton et al., 2015]), and MixUp augmentations ([Zhang et al., 2018]) and has been studied extensively in its own right ([Müller et al., 2019, Lukasik et al., 2020]). To obtain predictions during testing, the abnormality score is derived directly from the model's estimate of the interpolation factor α. Examples of A and A', with varying alpha, are shown in Figure 5.1. The corresponding label for each example is equal to the label mask scaled by the α value.

$$\alpha_i = \begin{cases} \alpha, & \text{if } i \in h \text{ and } A_i \neq B_i \\ 0, & \text{otherwise} \end{cases} \tag{5.5}$$

$$\mathcal{L}_{\text{bce}}(A', \alpha_i, f) = -\alpha_i \log(f(A')) - (1 - \alpha_i)\log(1 - f(A')) \tag{5.6}$$

$$\mathcal{L}_{\text{cce}}(A', \alpha_i, f) = -\sum_{c=1}^{N=5} \alpha_{i,c} \log(f(A')) \tag{5.7}$$

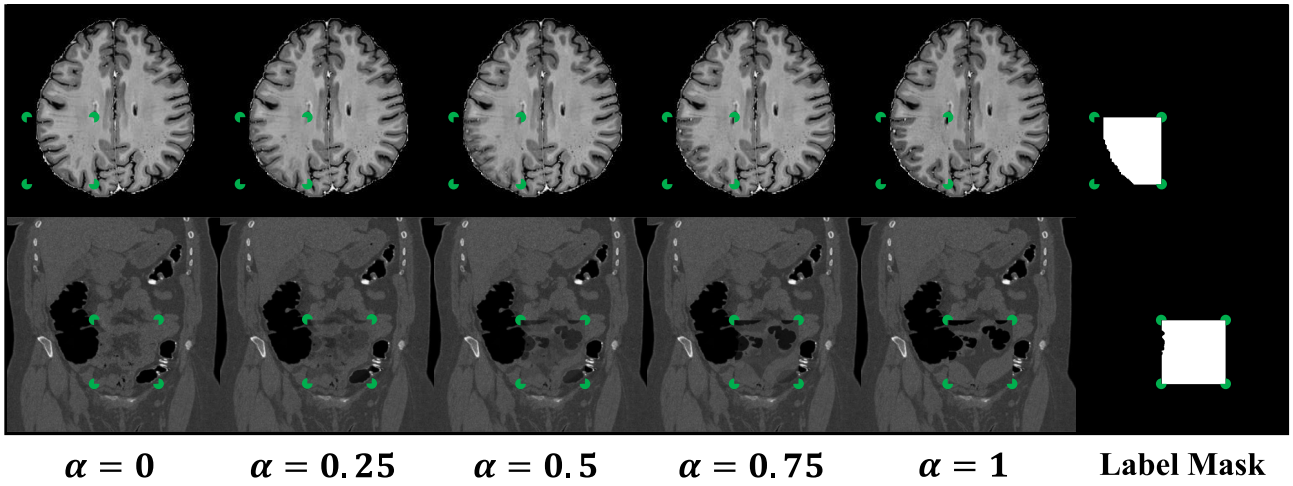


Figure 5.1: *Examples of foreign patch interpolation in brain and abdominal data from the MOOD challenge ([Zimmerer et al., 2020b]). Different α values correspond to different convex combinations. Scaling the label mask by the α value gives the label for each example. Green markers indicate the corners of the patch. Regions where A and A' are equal,* e.g., *background, are truncated in the label according to Eqn. 5.5. More examples are given in Appendix 5.A and 5.B.*

Note that FPI does not involve any image registration steps. Nevertheless, it is able to create a range of subtle training samples through simple linear interpolation (as seen in Figure 5.1 and Appendices 5.A and 5.B). We experiment on datasets with varying degrees of alignment, *e.g.*, brain MRI volumes with affine registration and CT data with no alignment (details in Section 5.4.1). In all cases, FPI is able to form useful training samples that improve detection of outliers.

Architecture

The network architecture is a wide residual encoder-decoder. The encoder portion is a standard wide residual network ([Zagoruyko and Komodakis, 2016]) with a width of 4 and a depth of 14. This is designed for inputs with dimensions 256x256. For inputs with dimensions 512x512, an additional residual block is added, bringing the depth up to 16. The decoder follows the same structure as the encoder but in reverse. The terminating activation is sigmoid in the case of continuous α or softmax with the appropriate number of output channels for discrete α.

Training

Training examples are created dynamically during training with random shuffling of the training data at the start of every epoch. This creates different convex combinations with different samples. Each model is trained for 50 epochs using Adam ([Kingma and Ba, 2015]) with a learning rate of 10^{-3}. An additional training phase can be performed after regular training for stochastic weight averaging ([Izmailov et al., 2018]). This step is not necessary to achieve good performance (Figure 5.4). However, we include its implementation details for completeness. Note that stochastic weight averaging was used in our submission to the MOOD challenge ([Zimmerer et al., 2020b]). To perform stochastic weight averaging, the model is trained for an additional 10 epochs with stochastic gradient descent ([Robbins and Monro, 1951]) and a cyclic learning rate oscillating in the range $[10^{-4}, 10^{-3}]$. The varying learning rate helps the model to escape minima and settle in new ones. The parameters are saved whenever the learning rate reaches a minimum (once per epoch). The final model is consolidated by taking the mean of the 10 saved minima. Stochastic weight averaging has been shown to give better generalization ([Izmailov et al., 2018]) and approximates ensembling methods without needing to increase model capacity.

5.4.1 Evaluation

Our method is evaluated on three datasets. The first two come from the MOOD challenge ([Zimmerer et al., 2020b]), while the third is a universal lesion dataset, DeepLesion ([Yan et al., 2018a, Yan et al., 2018b]).

MOOD Datasets ([Zimmerer et al., 2020b]): the MOOD challenge provides two datasets, 800 brain MRI volumes (256x256x256) and 550 abdominal CT volumes (512x512x512). Each subject is positioned in approximately the same way, but non-rigid registration is not used. As such, the same voxel/location in two different volumes may contain different tissue. All samples are assumed to be healthy with no abnormalities. Given that no test data is provided, we reserve 10% of the data as healthy test cases and we use 30% of the data to create anomalous test cases. The remaining 60% of the data is used for training. To create the anomalous test set, we synthesize five types of outliers. In each case a sphere of random size and location is selected within each volume; the pixels within that sphere are altered in one of five ways listed below. An example of a sink/source synthetic outlier is given in Figure 5.2. Performance is evaluated using average precision (AP), which is the metric originally used in the MOOD challenge ([Zimmerer et al., 2020b]). We also include evaluation with area under the receiver operating characteristic curve (AUROC) and an estimated DICE score ($\lceil$DICE$\rceil$). To compute an approximate DICE score, pixel-level anomaly scores are converted to binary segmentation masks. Following Baur et al., a greedy search is used to find an ideal threshold for this conversion ([Baur et al., 2021]).

- Uniform addition - a sphere of uniform intensity is added to the image;

$$A'_i = A_i + n, \ \forall \ i \in h, \ \text{where } n \sim \mathcal{N}(0,1) \tag{5.8}$$

- Noise addition - a sphere of random intensities is added to the image;

$$A'_i = A_i + n_i, \ \forall \ i \in h, \ \text{where } n_i \sim \mathcal{N}(0,1) \tag{5.9}$$

- Sink/source deformation - pixels are shifted toward/away from the center of the sphere;

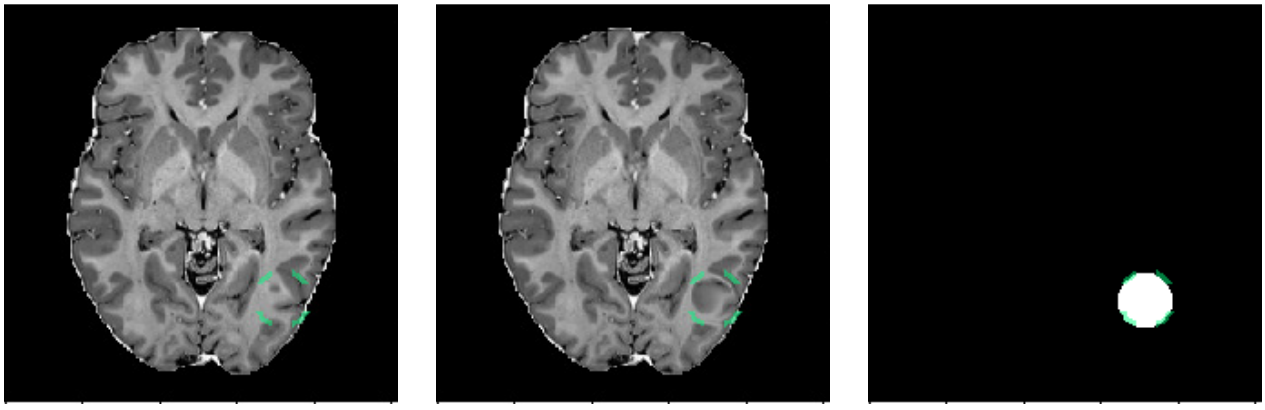
$$A'_I = A_V, \ \forall \ I \in h, \ \text{where } I = (i, j, k) \text{ and}$$
$$V = \begin{cases} h_c + s(I - h_c), & \text{for source} \\ I + (1 - s)(I - h_c), & \text{for sink} \end{cases} \tag{5.10}$$
$$\text{and } s = \left(\frac{\|I - h_c\|_2}{\frac{h_s}{2}} \right)^2$$

- Uniform shift - pixels in the sphere are resampled from a copy of the volume which has been shifted by a random distance in a random direction;

$$A'_{i,j,k} = A_{i+a,\, j+b,\, k+c} \ \forall \ i, j, k \in h,$$
$$\text{where } a, b, c \sim \sigma \mathcal{U}(0.02 \cdot d, 0.05 \cdot d) \tag{5.11}$$
$$\text{and } \sigma = \begin{cases} +1, & \text{with prob. } \frac{1}{2} \\ -1, & \text{with prob. } \frac{1}{2} \end{cases}$$

- Reflection - pixels in the sphere are resampled from a copy of the volume that has been reflected along an axis of symmetry.

$$A'_{i,j,k} = A_{i,d-j,k} \ \forall \ i, j, k \in h, \tag{5.12}$$
$$\text{where } d \text{ is image width}$$



(a) *Original* (b) *Synthetic Outlier* (c) *Label*

Figure 5.2: *Example of sink/source deformation used to synthesize an outlier. Original sample from MOOD challenge ([Zimmerer et al., 2020b]). All types of synthetic outliers are displayed in Appendix 5.C.*

DeepLesion Dataset ([Yan et al., 2018a]): this dataset contains CT scans from 4,427 unique patients exhibiting a broad range of lesions. There are at least eight different types of lesions including lung, abdomen, mediastinum, liver, pelvis, soft tissue, kidney, and bone. Each lesion is annotated with a bounding box. This dataset also includes volumetric data with slices above and below the annotated slice, typically about 30mm on both sides. In many cases, there are multiple annotated slices contained within one volume. To extract normal data from these volumes, we remove all annotated slices along with a margin about 10mm on either side. We train on 270,561 normal slices and test on 116,026 normal slices and 4831 annotated slices with lesions. A supervised benchmark is also trained using 22,496 slices with lesions and corresponding bounding box labels. Image-level testing uses normal slices and slices containing lesions. However, for pixel-level evaluation we only use slices with lesions. In this case, pixels inside bounding boxes are considered anomalous and all pixels outside of the bounding boxes are considered normal. All images are resized to 256x256. Performance is evaluated using AUROC and ⌈DICE⌉. Receiver operating characteristic (ROC) curves are also plotted.

The annotations in this dataset are mined from radiology reports and the creators of DeepLesion acknowledge that there may be lesions that have not been annotated, *e.g.*, those that were not relevant to the radiologist's examination ([Yan et al., 2018a]). While this makes training more difficult for outlier detection methods (and also for supervised methods), it represents a more realistic scenario, where it is difficult to ensure that the normal data contains no abnormalities of any kind.

5.4.2 Benchmark Methods

To evaluate the performance of the proposed method, foreign patch interpolation (FPI), we compare with several benchmark methods. For the MOOD challenge data with synthetic outliers, we compare with deep support vector data description (SVDD) ([Ruff et al., 2018]), a convolutional autoencoder (CAE) ([Masci et al., 2011]), and a maximum-mean discrepancy VAE (MMD-VAE) ([Zhao et al., 2019]). For the DeepLesion data with real medical abnormalities, we compare with several more advanced benchmarks including MMD-VAE ([Zhao et al., 2019]), a

hierarchical vector-quantized VAE (VQ-VAE2) ([Razavi et al., 2019b]), a restoration approach with VQ-VAE2 ([You et al., 2019, Marimont and Tarroni, 2021]), and a supervised method.

Deep SVDD ([Ruff et al., 2018]) is an embedding-based approach that learns a compact representation of the normal data. The network used is a convolutional encoder with equivalent depth to the encoder of FPI.

CAE ([Masci et al., 2011]) is a reconstruction-based method. It reconstructs images using features that are learned from normal data. Errors in the reconstruction are then used to highlight abnormal regions. The architecture for the CAE is a convolutional network with equivalent depth to the FPI network.

MMD-VAE ([Zhao et al., 2019]) uses maximum-mean discrepancy (MMD) ([Gretton et al., 2007]) to measure the distance between a prior and the distribution of encodings from real samples. Compared to conventional VAE's, this method is more stable during training and produces high fidelity reconstructions. For our implementation of MMD-VAE, we use the same wide residual encoder-decoder as FPI. Fully connected layers are added to the bottleneck resulting in latent codes of dimension 128.

VQ-VAE2 ([Razavi et al., 2019b]) compresses inputs by quantizing latent codes into discrete values at two levels of the network. We implement VQ-VAE2 using the same wide-residual encoder decoder network as FPI. Vector-quantization is performed at the two deepest layers (closest to the bottleneck). These have dimensions 32x32 and 16x16 respectively. At both levels, latent codes are quantized into 128 discrete values. The activations from the second deepest layer of the encoder are combined with the output of the first layer of the decoder. This skip connection structure allows VQ-VAE2 to produce more accurate reconstructions ([Razavi et al., 2019b]).

VQ-VAE2 Restoration ([You et al., 2019, Marimont and Tarroni, 2021]) uses two PixelCNN models ([Van Oord et al., 2016, Oord et al., 2016]), one at each of the vector quantized layers of the VQ-VAE2. Note that the second PixelCNN takes the latent codes from the first PixelCNN as a conditional input. After learning the distribution of the latent codes, the PixelCNN models

can be used to estimate the likelihood of each discrete code. Codes that are deemed to have a low likelihood are discarded and resampled from the learned distribution. The corrected codes are then used to produce a restored image and an anomaly scores is computed from the reconstruction error. Both PixelCNN models are composed of four residual blocks with masked convolutions and four masked convolutional layers on their own.

StyleGAN implementation of AnoGAN ([Karras et al., 2019, Schlegl et al., 2017]) is a reconstruction-based approach that aims to find a normal version of the query sample in the latent space of a GAN. In this case, a StyleGAN ([Karras et al., 2019]) is used. The model is trained from scratch at progressively higher resolutions, which improves stability and helps to produce more detailed, high resolution images. Instead of using a single latent code as input to the generator, StyleGAN maps a latent code into multiple style codes that are used to control adaptive instance normalization layers throughout the generator ([Huang and Belongie, 2017]). Gaussian noise is also added at different layers throughout the generator as a source of variation. To reconstruct a query image, we sample 80 initial sets of latent codes and noise vectors and find the set that gives the lowest reconstruction error. Then we further optimize the latent code and noise vectors to minimize the reconstruction error with 20 gradient steps.

Supervised training is also done for comparison. Unlike all other benchmarks, which are trained on only normal data, this supervised method is trained on only abnormal data. Lesion bounding boxes are used as labels. The network architecture is the same as the wide residual encoder decoder used in FPI. As such, this benchmark is trained in the same way as FPI, except the labels are real lesion bounding boxes rather than synthetic patch masks. Other more sophisticated supervised methods use region proposal networks to identify and classify patches ([Yan et al., 2018b]). But our arrangement allows us to directly assess the value of ground truth annotations compared with artificial labels generated by FPI.

5.5 Results

We first evaluate FPI on the MOOD challenge data and our synthetic testset. This includes an ablation study and comparison with simple baselines. Then we present results on the DeepLesion dataset and compare with more advanced benchmark methods.

5.5.1 MOOD Datasets with Synthetic Anomalies

Using the synthetic test data described in Section 5.4.1, we evaluate the method's ability to detect different types of outliers. Figure 5.3 displays the model's response to a sink/source deformation outlier and a normal sample. The plot includes abnormality scores for individual slices across the entire volume. Slices that include the artificially deformed sphere produce a strong and consistent activation (Figure 5.3, red). Meanwhile, normal slices elicit only weak activations (Figure 5.3, blue).

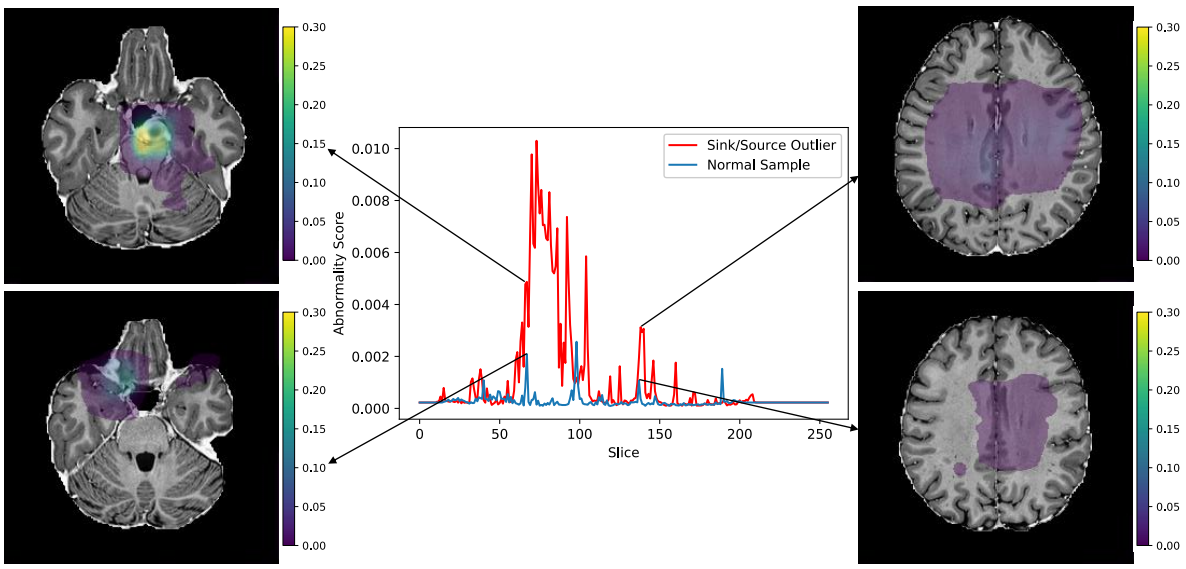


Figure 5.3: *Image-level abnormality scores for slices throughout the volume. Showing sink/source outlier (red plot and top images) and normal sample (blue line and bottom images). Data from MOOD ([Zimmerer et al., 2020b]). Slices with deformation have high anomaly scores (red), concentrated around the bulbous deformation (top left image). Normal sample has minimal abnormality scores.*

We perform an ablation study by modifying the self-supervision task. A 'binary' model is trained using a binary interpolation factor ($\alpha \in \{0, 1\}$). For a 'continuous round-up' model,

the training examples are generated using a continuous interpolation factor ($\alpha \in [0, 1]$), but the label supplied to the model is binary ($\alpha = 1$ if $\alpha > 0$). We also compare continuous and discrete configurations ($\alpha \in \{0, 0.25, 0.50, 0.75, 1\}$) as well as the application of stochastic weight averaging. Figure 5.4 displays the results for individual types of outliers and also overall sample and pixel level scores. Note that the overall scores (Figure 5.4, blue and green) are calculated using all outlier samples and all normal samples, so the class distribution is different from the individual scores. The binary and continuous round-up models are not able to detect the outliers in the test set effectively. Both continuous and discrete models achieve high performance, even without stochastic weight averaging. The low performance of the continuous stochastic weight averaged model may indicate that optimization is less stable for the continuous task. In contrast, stochastic weight averaging does not hurt performance for the discrete model and can substantially improve pixel-level scores.

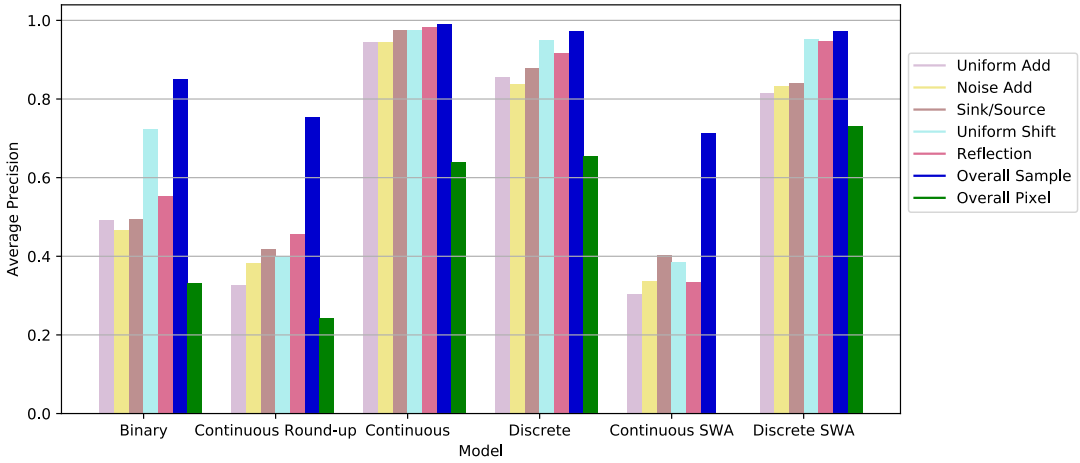


Figure 5.4: *Average precision for MOOD brain data ([Zimmerer et al., 2020b]) using different model configurations. The binary and continuous round-up models serve as simplified methods used in our ablation study. The continuous and discrete models represent our standard method. The addition of SWA is an optional extension.*

The abdominal models were trained in a similar manner and the discrete stochastic weight averaged model achieved the best overall performance. Table 5.1 shows the performance of the final selected models which are both trained using the discrete stochastic weight averaged configuration.

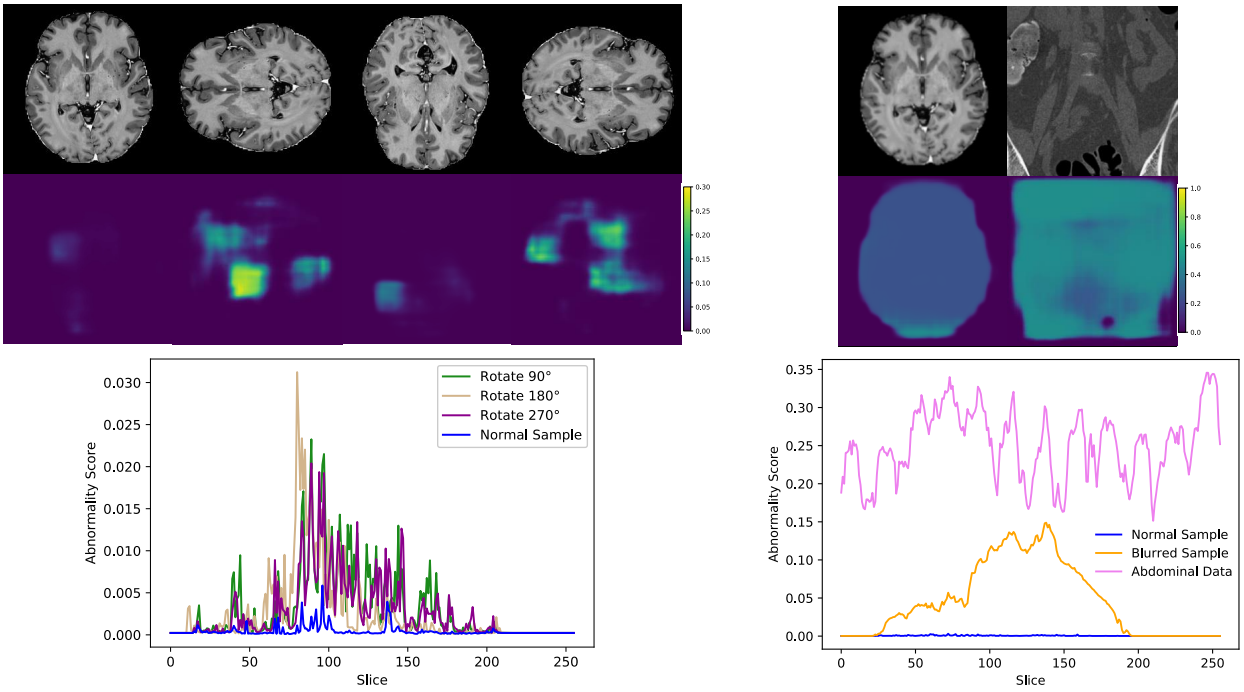
Since FPI is trained on synthetic examples, *i.e.*, interpolated patches, it is able to detect other similar classes of synthetic anomalies relatively easily. In comparison, reconstruction-

Table 5.1: Evaluation on synthetic test data, originally from brain and abdominal MOOD data ([Zimmerer et al., 2020b]).

Anatomy	Method	Subject-level		Pixel-level		
		AP	AUROC	AP	AUROC	⌈DICE⌉
Brain	Deep SVDD ([Ruff et al., 2018])	0.7695	0.5058	–	–	–
	CAE ([Masci et al., 2011])	0.7617	0.4947	0.0120	0.8695	0.0269
	MMD-VAE ([Zhao et al., 2019])	0.7572	0.4925	0.0144	0.8790	0.0350
	FPI (ours)	**0.9723**	**0.9321**	**0.7319**	**0.9852**	**0.7092**
Abdomen	Deep SVDD ([Ruff et al., 2018])	0.8318	0.5648	–	–	–
	CAE ([Masci et al., 2011])	0.7378	0.4717	0.0096	0.7240	0.0285
	MMD-VAE ([Zhao et al., 2019])	0.7356	0.4737	0.0079	0.7228	0.0235
	FPI (ours)	**0.8854**	**0.8025**	**0.6229**	**0.9292**	**0.6354**

based methods have difficulty identifying these synthetic anomalies because they have minimal intensity differences and occupy less than 1% of the total imaging volume of a subject. Although the reconstruction-based methods have high scores for pixel-level AUROC, the DICE scores are quite low (Table 5.1). This is because the DICE score focuses more on anomalous pixels, while AUROC can be partly inflated by a large number of normal background pixels. These blank pixels are easy to reconstruct without error. This increases the number of true negatives, which in turn decreases the false positive rate (x-axis of the ROC curve) and increases the area under the curve. In contrast, pixels in tissue regions often have some level of reconstruction error because there is a limit to the amount of detail that the models can recreate. Since the synthetic anomalies have similar intensity values, they also produce similar reconstruction error. When the reconstruction error is averaged across the entire volume, the contribution from the synthetic anomaly is hidden by the contributions from other healthy regions, which leads to a poor subject-level AUROC. Meanwhile, FPI produces very low anomaly scores for normal tissue and activates specifically for certain types of features, as seen in Figure 5.3.

In addition to the synthetic test set, which only includes local abnormalities, we provide examples of global abnormalities in Figure 5.5. A normal sample produces minimal activation in its canonical orientation (Figure 5.5, left most image in (a)). However, rotating the sample produces scattered activations throughout the entire volume (Figure 5.5, (a)). Blurring or substituting different anatomy produces even stronger activations (Figure 5.5, (b)).



(a) *Rotations* (b) *Blurred and abdominal data*

Figure 5.5: *Examples of global outliers using MOOD data ([Zimmerer et al., 2020b]). (a) Original normal sample (top left) and rotations. (b) Gaussian blur ($\sigma = 1$) and abdominal data. Note the change of scale in activation maps. Plots display the abnormality score across slices.*

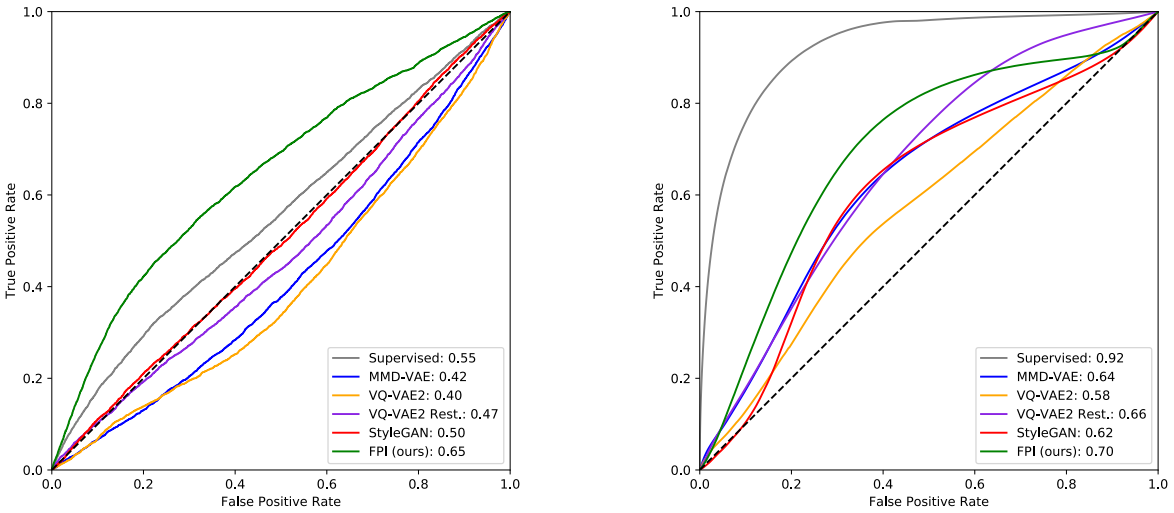
5.5.2 DeepLesion Dataset with Medical Anomalies

For the DeepLesion dataset, FPI was trained under the continuous α (interpolation factor) setting without stochastic weight averaging. The results demonstrate that FPI can identify real medical anomalies despite being trained on only normal images. Table 5.2 displays both image and pixel level AUROC scores as well as estimated DICE scores. ROC curves are shown in Figure 5.6.

At the image level, the reconstruction-based methods score around 0.5 or below. Several factors contribute toward this, including high variation in normal data, higher reconstruction error from certain structures, and overrepresentation of certain tissue types in the normal test data. Figure 5.8 shows that reconstruction-based models must learn to reproduce a wide range of structures and different organs. Most of the reconstruction error comes from sharp edges with high contrast and high spatial frequency, *i.e.*, tissue interfaces. Also, the more pixels involved, the higher the contribution to the overall (image-level) anomaly score. As an example, the lungs

Table 5.2: Evaluation on DeepLesion data ([Yan et al., 2018a]). Image-level evaluation is performed using normal slices and slices with lesions. Pixel-level evaluation is done using only slices with lesions; bounding boxes serve as approximate lesion segmentation masks.

Method	Image-level AUROC	Pixel-level AUROC	Pixel-level ⌈DICE⌉
Supervised	0.554	0.923	0.226
MMD-VAE ([Zhao et al., 2019])	0.419	0.635	0.024
VQ-VAE2 ([Razavi et al., 2019b])	0.405	0.576	0.018
VQ-VAE2 Restoration ([You et al., 2019, Marimont and Tarroni, 2021])	0.469	0.664	0.023
StyleGAN ([Karras et al., 2019])	0.501	0.618	0.023
FPI (ours)	**0.648**	**0.701**	**0.030**



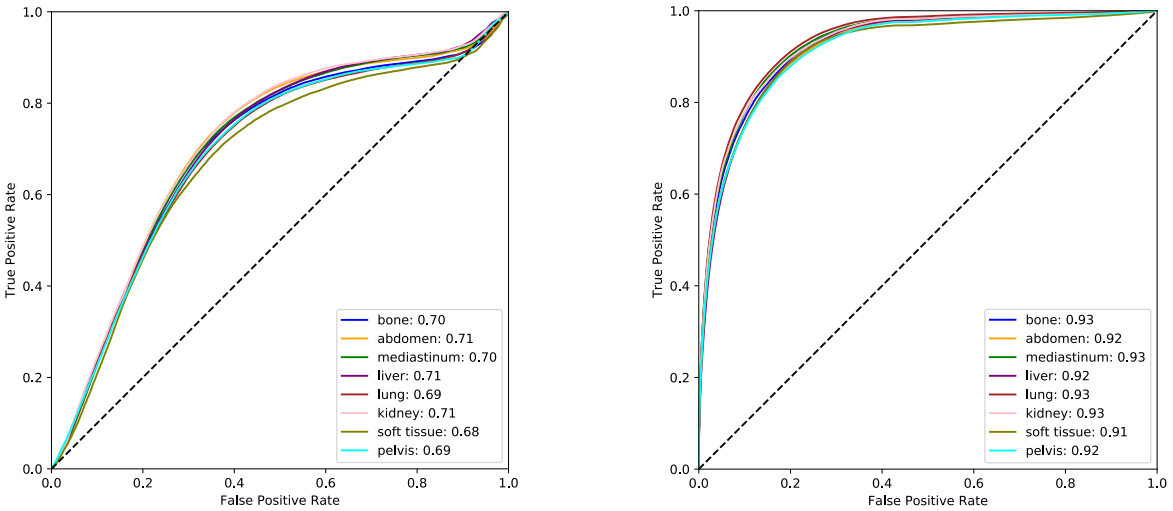
(a) *Image-level* (b) *Pixel-level*

Figure 5.6: *ROC curves for DeepLesion data ([Yan et al., 2018a]) for each method at the image-level (a) and pixel-level (b). AUROC reported in the legend.*

generally have a high reconstruction error because they span across a large area and contain details with high spatial frequency. The lungs may also be overrepresented in the normal test data. As described in Section 5.4.1, each anomalous test image is accompanied by parallel slices that give context above and below the anomalous slice. The context slices, minus a margin around the anomalous slice, are used as normal test data, resulting in 116,026 normal test images and 4831 anomalous test images. However, certain regions have more context slices than others. For example, the average number of context slices for an anomalous lung image is 79, whereas soft tissue type lesions (muscle, skin, fat) only have 37 context slices on average. As such, the normal test data may be skewed toward certain organs that have high reconstruction

error. This can increase the false positive rate and reduce the area under the ROC curve. This skew may exist in the training data as well, but reproducing details with high spatial frequency can still be challenging for methods that rely on a lower dimensional representation of the data.

The supervised method, which is only trained on slices containing lesions, also performs poorly when tested on images that are both normal and abnormal. This could be fixed by including normal samples during supervised training. But it illustrates that even supervised methods can face difficulty when the test distribution does not match the training distribution. FPI is specifically designed to handle out-of-distribution samples and does not rely on proxy tasks that require full image reconstruction. These properties makes it suitable for detecting subtle lesions within highly variable data.



(a) *FPI by lesion type* (b) *Supervised by lesion type*

Figure 5.7: *Pixel-level ROC curves for individual lesion types of DeepLesion data ([Yan et al., 2018a]). FPI and a supervised method are plotted in (a) and (b), respectively.*

For the pixel-level score, only slices with lesions are considered so that we can directly assess localization. The supervised method excels in this setting because the training and test data are consistent. Even so, the supervised DICE score is modest and the others are quite low. This can be partly attributed to the fact that bounding boxes are used as approximate segmentation masks. Although the pixel-level anomaly predictions may not overlap accurately with the complete bounding boxes, the AUROC scores indicate that these regions tend to be rated as more anomalous. This level of performance is insufficient for lesion segmentation, but may be reasonable for highlighting suspicious regions in an anomaly setting. All unsupervised methods

achieve an AUROC over 0.5 with FPI scoring the highest among the unsupervised methods. Full ROC curves are plotted in Figure 5.6 (b). Individual ROC curves for each lesion type are also shown for FPI in Figure 5.7 (a) and for the supervised method in Figure 5.7 (b). FPI performs similarly on each lesion type, indicating that it is equally sensitive to a broad range of lesions.

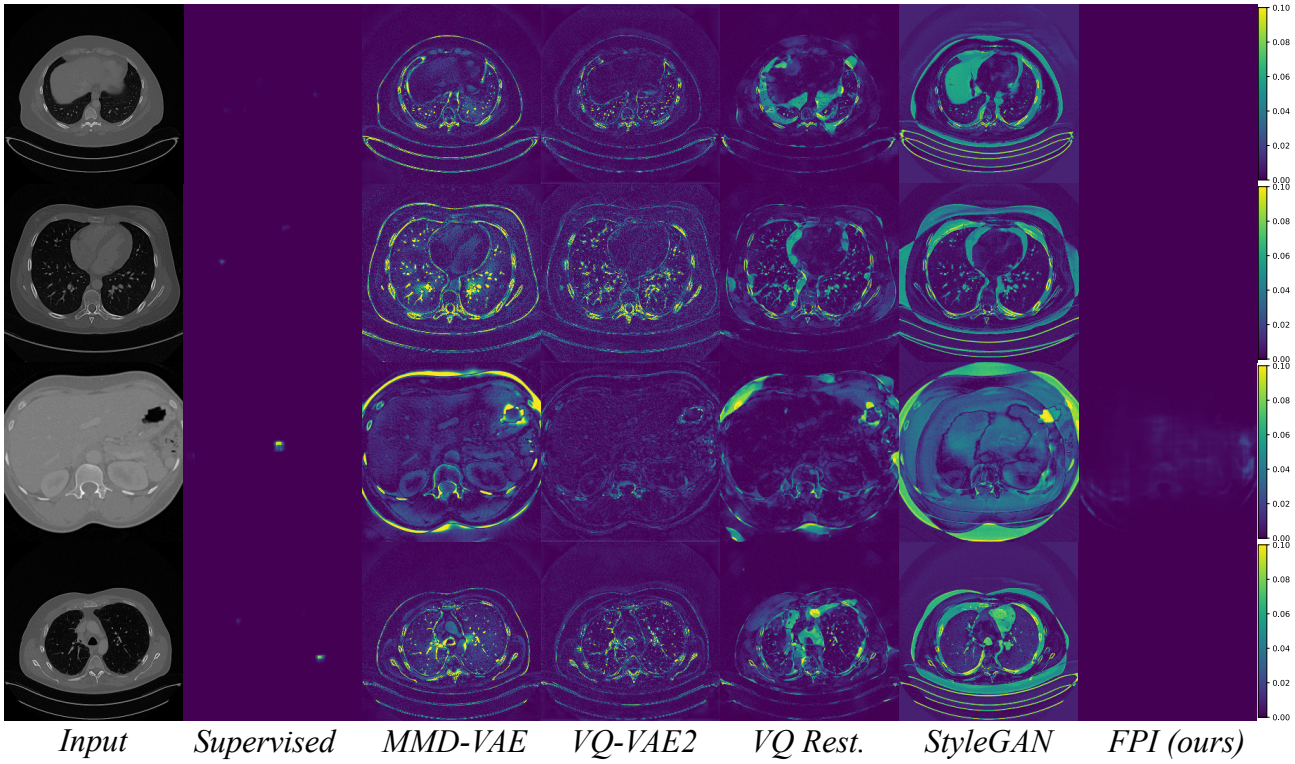


Figure 5.8: *Normal test samples from DeepLesion ([Yan et al., 2018a]) and outputs from each method. Note that reconstruction error outputs are scaled down by a factor of five.*

Figure 5.9 displays anomalous examples from the DeepLesion dataset with bounding box labels for each lesion. The outputs from each method show varying levels of sensitivity. MMD-VAE exhibits reconstruction errors throughout the images which reflects the difficulty of learning a compact representation for data with high variation and detail. VQ-VAE2 uses a hierarchical architecture to produce higher fidelity reconstructions with less error. However, this does not help the network to be sensitive to specific irregularities such as lesions. Using the VQ-VAE2 for image restoration can help to highlight regions based on likelihood, rather than purely on intensity differences. This approach can be more selective, but it also tends to highlight certain natural variations that may be deemed less likely. Meanwhile, StyleGAN searches for a normal

matching image in its latent space, but it is not always possible to find a good match when the data has complex and detailed structures that can vary greatly across images. In comparison to the reconstruction-based methods, FPI highlights more specific areas in the image that contain lesions or other unusual elements that are not lesions. Finally, the supervised method gives the most lesion-specific activations which can only be learned through labelled examples.

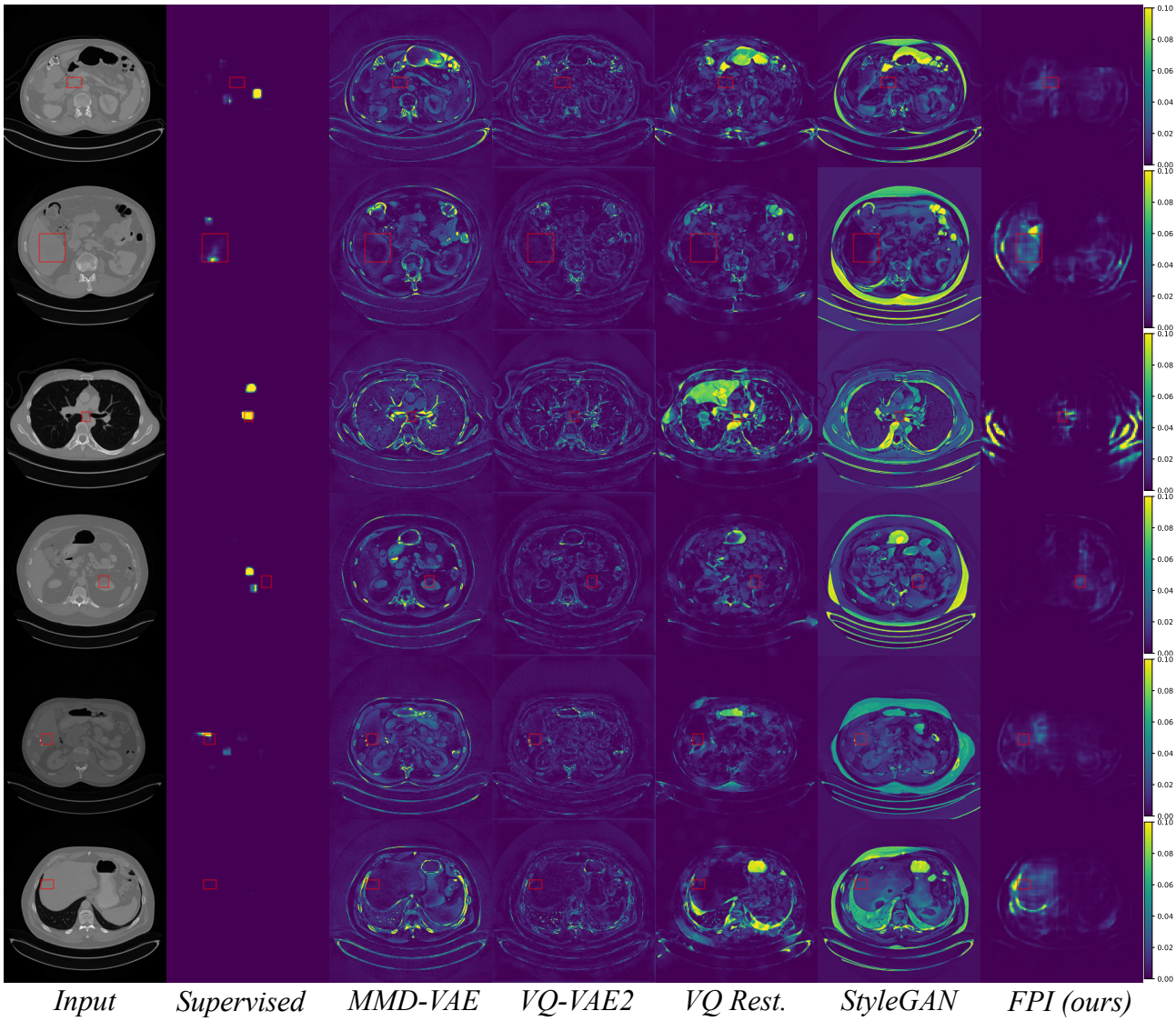


Figure 5.9: *Anomalous test samples from DeepLesion ([Yan et al., 2018a]) and outputs from each method. Bounding boxes indicate lesions. Note that reconstruction error outputs are scaled down by a factor of five.*

5.6 Discussion

The proposed method uses a simple self-supervised task to simulate subtle irregularities in the image. Our ablation study suggests that two aspects of this task are important, exposure and difficulty. Without these qualities, the network can overfit to the self-supervised task and fail to detect other types of anomalies. For instance, generating samples with a binary interpolation factor limits the network's exposure to samples with swapped patches. This leads to poor generalization to other types of synthetic anomalies (Figure 5.4, 'Binary'). A varying interpolation factor provides exposure to abnormalities with varying levels of subtlety. However, exposure is not sufficient on its own. The challenge of estimating the *value* of the interpolation factor is also crucial. If training examples are created using a varying interpolation factor ($\alpha \in [0, 1]$), but the task is simplified by rounding the label to a binary value ($\alpha = 1$ if $\alpha > 0$) then generalization is also poor (Figure 5.4, 'Continuous Round-up'). The difficulty and variety of the proposed task allow FPI to achieve high performance, whether using continuous or discrete α values. Stochastic weight averaging can also provide some benefit, particularly in pixel-wise scores on our synthetic test data (Figure 5.4). Nonetheless, it is not strictly necessary and good results can be achieved without it.

Due to the nature of the self-supervised task and the synthesized outliers, one concern is that the network may only detect artifacts, such as discontinuities in image intensity. Indeed, if the characteristics of the synthetic anomalies are more consistent than the characteristics of the normal data, then the network may learn to recognize these artifacts instead of learning the normal appearance of healthy anatomy, which is the real goal. As such, we evaluate FPI using a range of synthetic anomalies, including intensity shifts and deformations; global anomalies that have no discontinuities; and real medical anomalies. The results demonstrate that FPI can detect a broad range of abnormalities, even if there are no discontinuities. This implies that the self-supervised task helps the network to learn the normal appearance of anatomy to some extent. Any deviations from that expectation are therefore seen as foreign patterns being introduced ($\alpha > 0$).

A major difference between this work and reconstruction-based methods is that we focus on

subtle irregularities. In a reconstruction-based approach, the abnormality score is directly proportional to the intensity differences between the test image and its reconstruction. This makes it difficult to detect more subtle irregularities, especially if the normal data has a high variance and is more difficult to faithfully reconstruct. The DeepLesion dataset exhibits both of these characteristics. The lesions can be very subtle and the anatomy varies considerably. In some cases the field of view is centered on the anatomy of interest and other structures are missing or misaligned. Our evaluation on the DeepLesion dataset indicates that reconstruction-based methods are sensitive to gross intensity differences and variations in anatomy. They are largely unable to selectively highlight subtle lesions (Figure 5.9). Image level AUROC for both reconstruction-based methods is actually below 0.5 (Table 5.2). This means that reconstruction error is higher in some normal slices than it is in abnormal slices. This could be because normal slices are peripheral to the lesion slices and may have more variance in structure. This in turn can raise the reconstruction error which is dominated by larger structural differences in the image. In contrast, FPI is able to ignore most variations in normal anatomy. Rather than trying to reconstruct every detail, FPI is trained to detect only regions that are incongruous with the rest of the image (*i.e.*, foreign patches). This allows FPI to be more sensitive to subtle irregularities such as lesions. In this way, FPI can complement reconstruction-based methods and detect less obvious cases that might otherwise require more intense scrutiny.

One challenge in unsupervised outlier detection is selecting the best model. Validation sets can be used to select the most performant model. However, this may introduce a bias toward the types of outliers in the validation set. Even if the validation set is disjoint from the test set, there are likely similarities. This may lead to overestimation of performance and failure on unexpected outliers encountered during deployment. As such, we avoid using outliers for validation and simply keep the training duration fixed. Using the same training regime we demonstrate FPI's capability across several datasets. For real world deployment, it may be important to add elements such as uncertainty estimation to make predictions more informative.

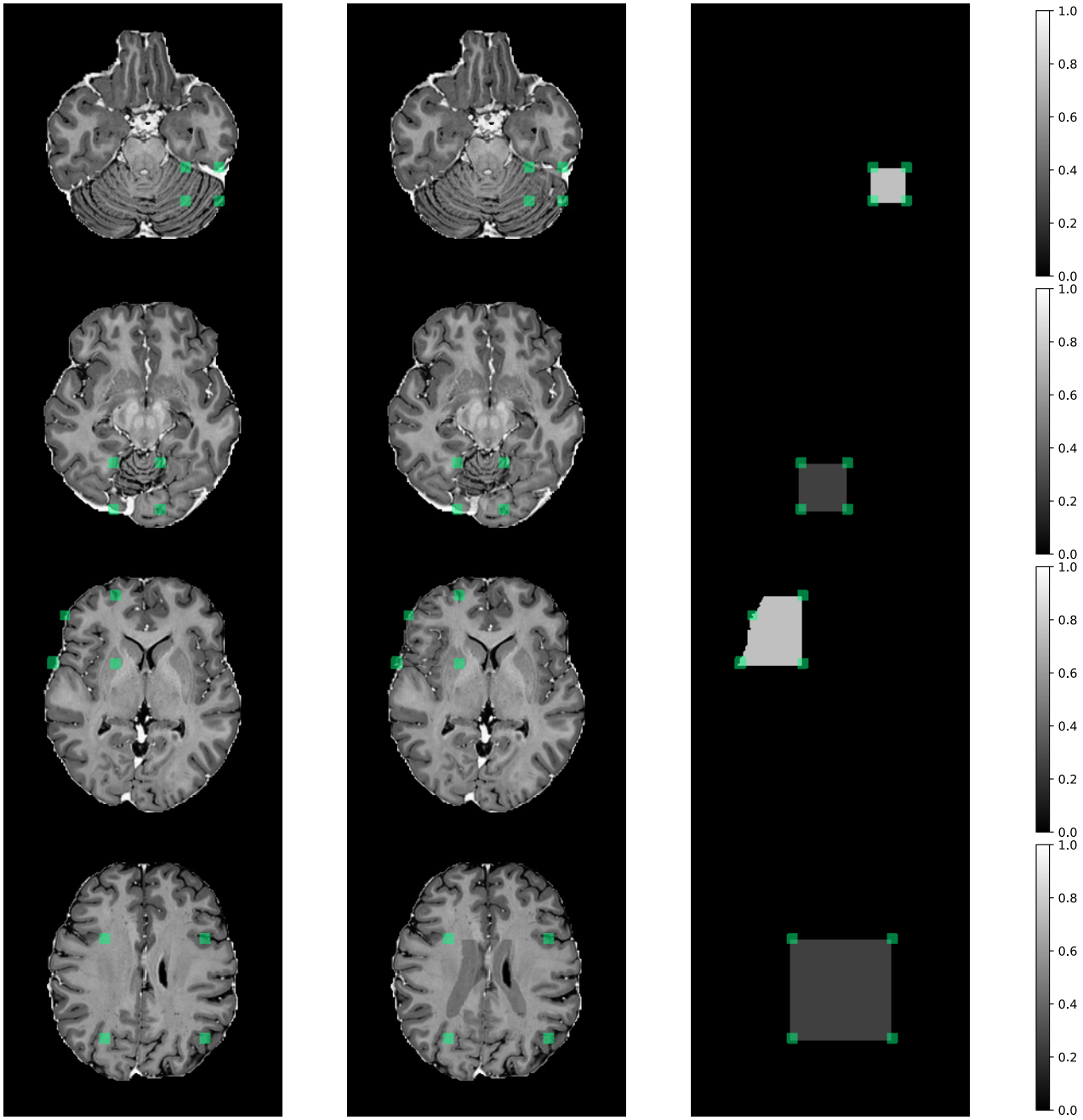
5.7 Summary

We propose a self-supervision framework for detecting fine-grained abnormalities, common in medical data. Foreign patterns are drawn from independent subjects and used to simulate abnormalities. The network is trained to detect where and to what degree a foreign pattern has been introduced. The resulting model is able to generalize to a wide range of subtle irregularities and achieved the highest rank in the 2020 MICCAI MOOD challenge ([Zimmerer et al., 2020b]) in both sample and pixel level tasks. We also demonstrate FPI's ability to detect a broad range of real medical lesions in the challenging DeepLesion dataset.

The goal of future work is to improve performance on cases where there is less structural consistency. Further extensions could also provide uncertainty estimates for the predicted anomaly scores. Ultimately we hope to reduce the burden placed on radiologists.

Code is available at `https://github.com/jemtan/FPI`.

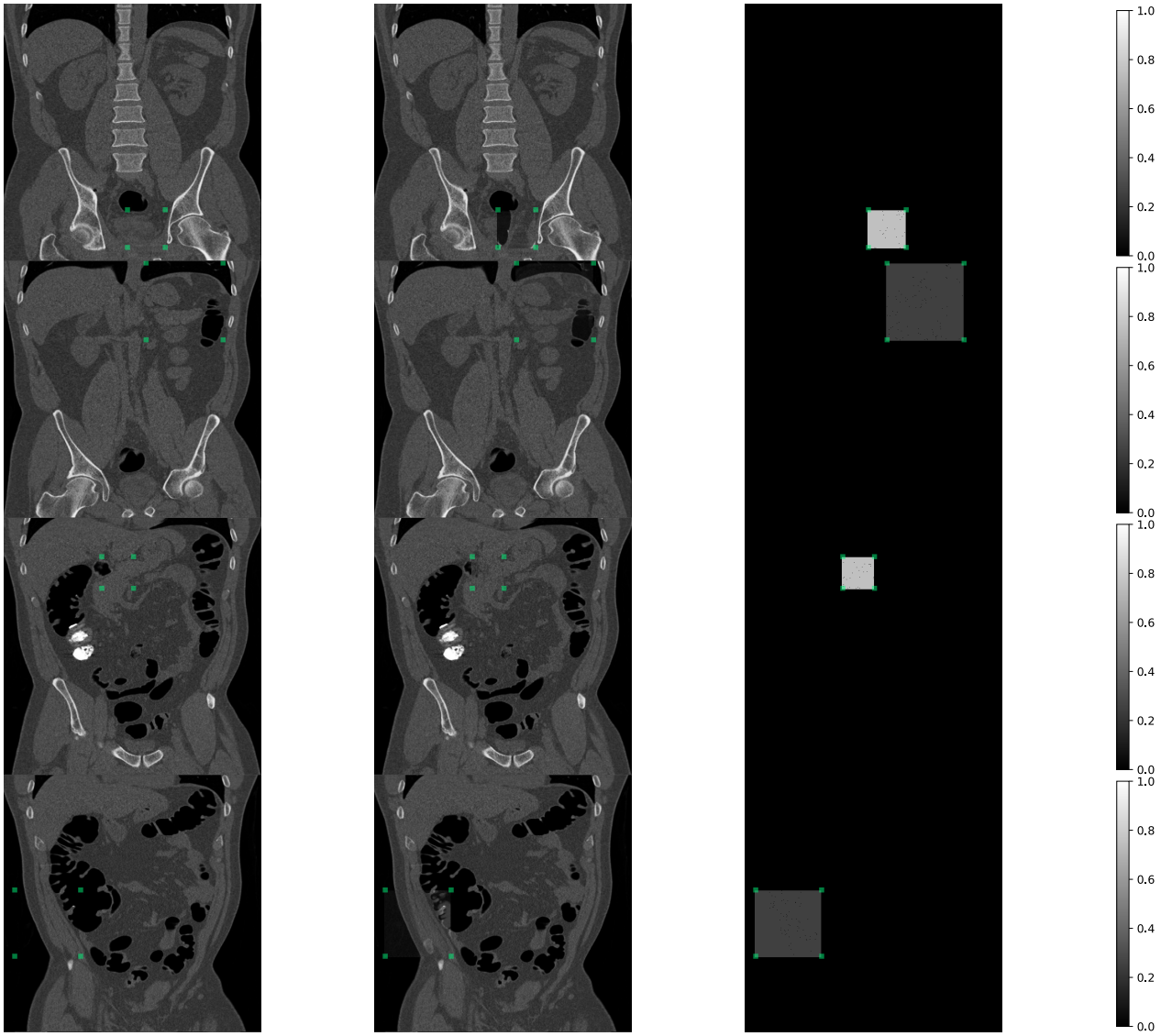
5.A FPI in Brain Images



(a) *Original* (b) *Interpolation* (c) *Label*

Figure 5.10: *MOOD brain images ([Zimmerer et al., 2020b]) with foreign patches.*

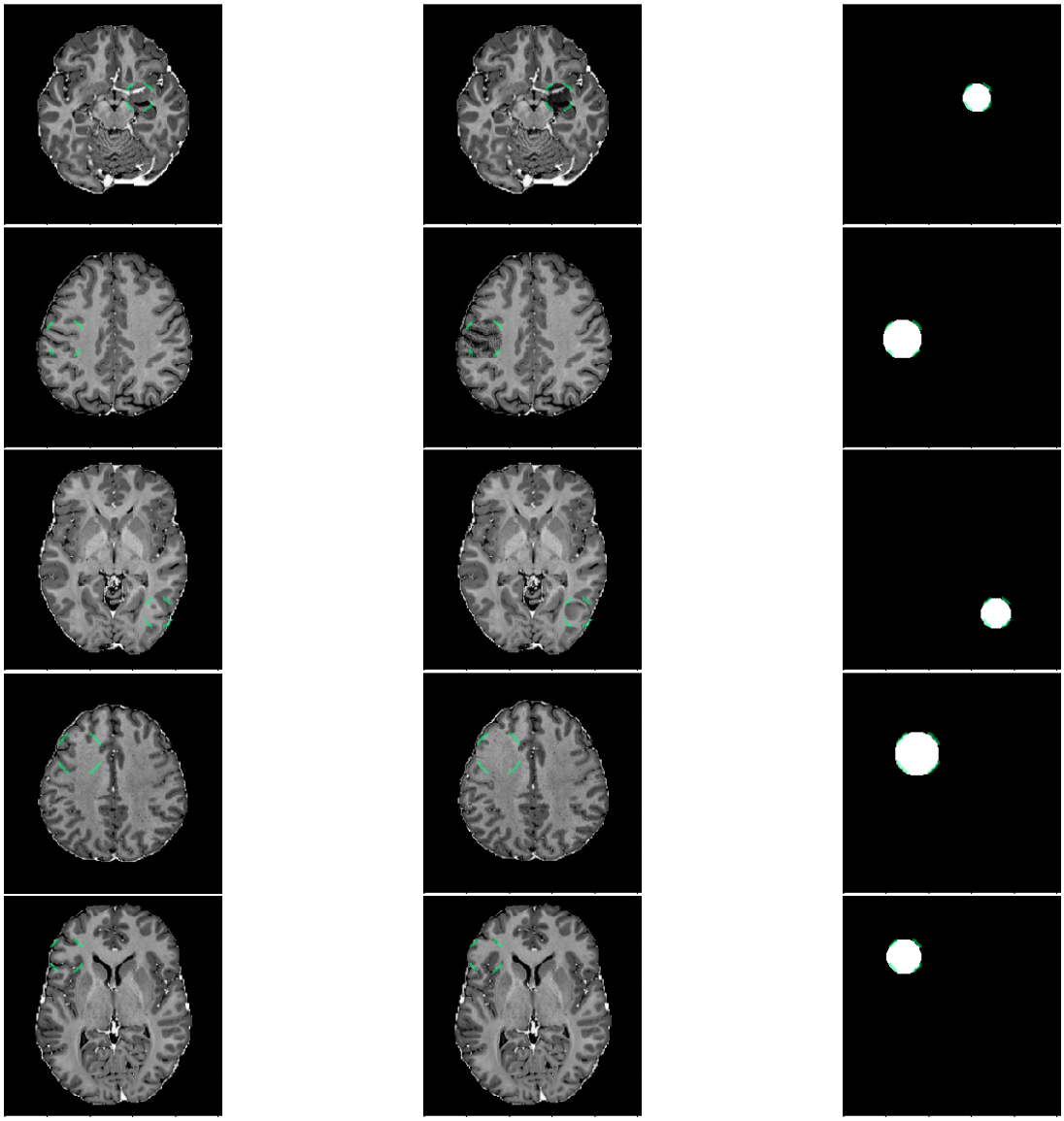
5.B FPI in Abdominal Images



(a) *Original* (b) *Interpolation* (c) *Label*

Figure 5.11: *MOOD abdominal images ([Zimmerer et al., 2020b]) with foreign patches.*

5.C Examples of Synthetic Outliers



(a) *Original* (b) *Synthetic Outlier* (c) *Label*

Figure 5.12: *Each row shows one type of synthetic outlier. From top to bottom these are uniform addition, noise addition, sink/source deformation, uniform shift, and reflection. Original data from MOOD challenge ([Zimmerer et al., 2020b]).*

Chapter 6

Enhancing Self-Supervised Outlier detection with Advanced Image Editing and Meta-Learning

Publications Associated with this Chapter

[1] Tan, J., Hou, B., Day, T., Simpson, J., Rueckert, D., and Kainz, B. (2021a). Detecting outliers with poisson image interpolation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 581–591. Springer

[2] Tan, J., Kart, T., Hou, B., Batten, J., and Kainz, B. (2021b). Metadetector: Detecting outliers by learning to learn from self-supervision. In *Biomedical Image Registration, Domain Generalisation and Out-of-Distribution Analysis at MICCAI*, pages 119–126. Springer

This chapter covers two methods that build on the work presented in the last chapter (foreign patch interpolation).

The first is Poisson image interpolation [Tan et al., 2021a], which enhances the self-supervised task with advanced image editing. In particular, this method improves the way that foreign patches are blended into the image by using Poisson image interpolation [Pérez et al., 2003]. This removes certain artifacts, such as sharp discontinuities, and broadens the range of possible applications.

The second part covers MetaDetector [Tan et al., 2021b], which incorporates some of the ideas from Chapter 4 into the self-supervised framework. The motivation behind MetaDetector is to learn more general features and allow for test time adaptation through meta-learning. MetaDetector was also submitted to the 2021 medical out-of-distribution analysis (MOOD) challenge [Petersen et al., 2021] and achieved second place in the pixel-level category.

6.1 Detecting Outliers with Poisson Image Interpolation

6.1.1 Overview

Supervised learning of every possible pathology is unrealistic for many primary care applications like health screening. Image anomaly detection methods that learn normal appearance from only healthy data have shown promising results recently. We propose an alternative to image reconstruction-based and image embedding-based methods and propose a new self-supervised method to tackle pathological anomaly detection. Our approach originates in the foreign patch interpolation (FPI) strategy that has shown superior performance on brain MRI and abdominal CT data. We propose to use a better patch interpolation strategy, Poisson image interpolation (PII), which makes our method suitable for applications in challenging data regimes. PII outperforms state-of-the-art methods by a good margin when tested on surrogate tasks like identifying common lung anomalies in chest X-rays or hypo-plastic left heart syndrome in prenatal, fetal cardiac ultrasound images.

6.1.2 Introduction

Doctors such as radiologists and cardiologists, along with allied imaging specialists such as sonographers shoulder the heavy responsibility of making complex diagnoses. Their decisions often determine patient treatment. Unfortunately, diagnostic errors lead to death or disability almost twice as often as any other medical error [Tehrani et al., 2013]. In spite of this, the medical imaging workload has continued to increase over the last 15 years [Bruls and Kwee,

2020]. For instance, on-call radiology, which can involve high-stress and time-sensitive emergency scenarios, has seen a 4-fold increase in workload [Bruls and Kwee, 2020].

One of the major goals of anomaly detection in medical images is to find abnormalities that radiologists might miss due to excessive workload or inattention blindness [Drew et al., 2013]. Most of the existing, automated methods are only suitable for detecting gross differences that are highly visible, even to observers without medical training. This undermines their usefulness in routine applications. Detecting anomalies at the level of medical experts typically requires supervised learning. This has been achieved for specific applications such as breast cancer [Wu et al., 2019] or retinal disease [De Fauw et al., 2018]. However, detecting arbitrary irregularities, without having any predefined target classes, remains an unsolved problem.

Recently, self-supervised methods have proven effective for unsupervised learning [Oord et al., 2018, Hénaff et al., 2019]. Some of these methods use a self-supervised task that closely approximates the target task [Chen et al., 2020b, He et al., 2020] (albeit without labels). There are also self-supervised methods that closely approximate the task of outlier detection. For example, foreign patch interpolation (FPI) trains a model to detect foreign patterns in an image [Tan et al., 2022]. The self-supervised task used for training takes a patch from one sample and inserts it into another sample by linearly interpolating pixel intensities. This creates training samples with irregularities that range from subtle to more pronounced. But for data with poor alignment and varying brightness, FPI's linear interpolation will lead to patches that are clearly incongruous with the rest of the image. This makes the self-supervised task too easy and reduces the usefulness of the learned features.

Contribution: We propose Poisson image interpolation (PII), a self-supervised method that trains a model to detect subtle irregularities introduced via Poisson image editing [Pérez et al., 2003]. We demonstrate the usefulness of PII for anomaly detection in chest X-ray and fetal ultrasound data. Both of these are challenging datasets for conventional anomaly detection methods because the normal data has high variation and outliers are subtle in appearance.

Related Work: Reconstruction-based outlier detection approaches can use autoencoders, variational autoencoders (VAEs) [Zimmerer et al., 2019b], adversarial autoencoders (AAEs) [Chen

and Konukoglu, 2018], vector quantised variational autoencoders (VQ-VAE) [Razavi et al., 2019a], or generative adversarial networks (GANs) [Schlegl et al., 2019]. Some generative models are also used for pseudo-healthy image generation [Xia et al., 2020]. Reconstruction can be performed at the image [Baur et al., 2021], patch [Wei et al., 2018], or pixel [Alaverdyan et al., 2020] level. In each case, the goal is to replicate test samples as closely as possible using only features from the distribution of normal samples [Baur et al., 2021]. Abnormality is then measured as intensity differences between test samples and their reconstructions. Unfortunately, raw pixel differences lack specificity, making semantic distinctions more difficult.

Disease classifiers specialize in making fine semantic distinctions. They do this by learning very specific features and ignoring irrelevant variations [LeCun et al., 2015]. To harness the qualities that make classifiers so successful, some methods compare samples as embeddings within a learned representation. For example, deep support vector data description (SVDD) learns to map normal samples to a compact hypersphere [Ruff et al., 2018]. Abnormality is then measured as distance from the center of the hypersphere. Other methods, such as [Marimont and Tarroni, 2021], learn a latent representation using a VQ-VAE and exploit the autoregressive component to estimate the likelihood of a sample. Furthermore, components of the latent code with low likelihood can be replaced with samples from the learned prior. This helps to prevent the model from reconstructing anomalous features. A similar approach has also been proposed using transformers [Pinaya et al., 2021]. Overall, comparing samples in a learned representation space can allow for more semantic distinctions. But with only normal training examples, the learned representation may emphasize irrelevant features, *i.e.*, those pertaining to variations *within* the normal class. This often requires careful calibration for applications.

Self-supervised methods aim to learn more relevant representations by training on proxy tasks. One of the most effective strategies is to train a classifier to recognize geometric transformations of normal samples [Golan and El-Yaniv, 2018, Tack et al., 2020]. By classifying transformations, the network learns prominent features that can act as reliable landmarks. Outliers that lack these key features will be harder to correctly classify when transformed. The anomaly score is thus inversely proportional to the classification accuracy. This works well for natural images, but in medical applications, disease appearance can be subtle. Many outliers still contain all

of the major anatomical landmarks.

To target more subtle abnormalities, some methods use a localized self-supervised task. For example, FPI synthesizes subtle defects within random patches in an image [Tan et al., 2022]. The corresponding pixel-level labels help the network to learn which regions are abnormal given the surrounding context. This approach showed good performance for spatially aligned brain MRI and abdominal CT data [Zimmerer et al., 2020a]. CutPaste [Li et al., 2021] also synthesizes defects by translating patches within an image. This is effective for detecting damage or manufacturing defects in natural images [Li et al., 2021]. However, unlike cracks or scratches seen in manufacturing, many medical anomalies do not have sharp discontinuities. Overfitting to obvious differences between the altered patch and its surroundings can limit generalization to more organic and subtle outliers. We propose to resolve this issue using Poisson image editing [Pérez et al., 2003]. This helps to create more subtle defects (for training) which in turn improves generalization to real abnormalities.

6.1.3 Method

To begin, we provide a brief description of FPI. Consider two normal training samples, x_i and x_j, of dimension $N \times N$, as well as a random patch h, and a random interpolation factor $\alpha \in [0, 1]$. FPI replaces the pixels in patch h with a convex combination of x_i and x_j, to produce a training image $\widetilde{x}_i$ (Eqn. 6.1). Note that $\widetilde{x}_i = x_i$ outside of h. For a given training image $\widetilde{x}_i$, the corresponding label is $\widetilde{y}_i$, as specified by Eqn. 6.2.

$$\widetilde{x}_{i_p} = (1 - \alpha)x_{i_p} + \alpha x_{j_p} \ , \ \forall \ p \in h \tag{6.1}$$

$$\widetilde{y}_{i_p} = \begin{cases} \alpha & \text{if } p \in h \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

This approach has similarities to mixup [Zhang et al., 2018], a data augmentation method that generates convex combinations of images and their respective labels. In the case of FPI, the training data only contains normal samples. Without having any class labels, FPI calculates its

own labels as convex combinations of self (0 for y_i) and non-self (1 for y_j) as shown in Eqn. 6.2.

If x_i and x_j have vastly different intensity levels or structures, the interpolated patch will be inconsistent with the rest of the image. These differences are easy to spot and provide no incentive for the model to learn features that constitute "normal" (a much harder task). To create more challenging cases, we use a technique for seamless image blending. Poisson image editing [Pérez et al., 2003] blends the content of a source image (x_j) into the context of a destination image (x_i). Rather than taking the raw intensity values from the source, we extract the relative intensity differences across the image, *i.e.* the image gradient. Combining the gradient with Dirichlet boundary conditions (at the edge of the patch) makes it possible to calculate the absolute intensities within the patch. This is illustrated in Figure 6.1.

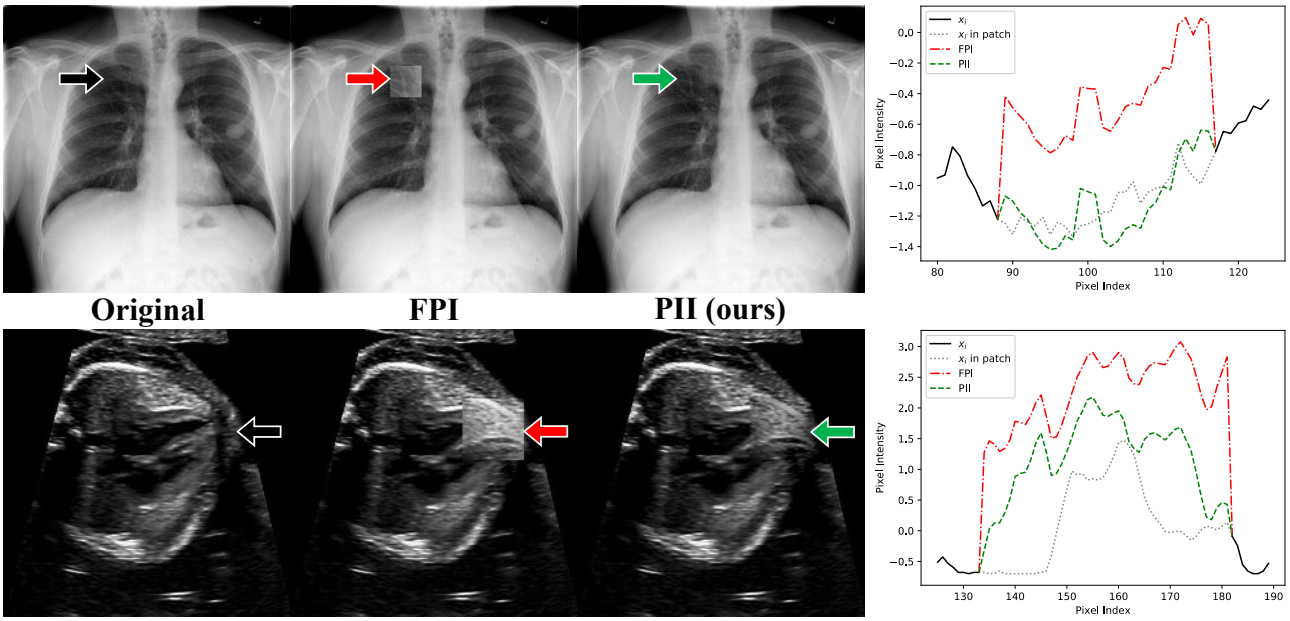


Figure 6.1: *Examples of patches altered by FPI (convex combination) and PII (Poisson blending). Arrows in the images indicate the location of the line plotted on the right. Altering patches can simulate subtle (top) or dramatic (bottom) changes to anatomical structures. In both cases, PII blends the changes into the image more naturally.*

More formally, let f_{in} be a scalar function representing the intensity values within the patch h. The goal is to find intensity values of f_{in} that will:

1. match the surrounding values, f_{out}, of the destination image, along the border of the patch (∂h), and

2. follow the relative changes (image gradient), $\mathbf{v}$, of the source image.

$$\min_{f_{in}} \iint_{h} |\nabla f_{in} - \mathbf{v}|^2 \text{ with } f_{in}\Big|_{\partial h} = f_{out}\Big|_{\partial h} \tag{6.3}$$

$$\Delta f_{in} = \text{div}\mathbf{v} \text{ over } h, \text{ with } f_{in}\Big|_{\partial h} = f_{out}\Big|_{\partial h} \tag{6.4}$$

These conditions are specified in Eqn. 6.3 [Pérez et al., 2003] and its solution is the Poisson equation (Eqn. 6.4). Intuitively, the Laplacian, ($\Delta\cdot = \frac{\partial^2 \cdot}{\partial x^2} + \frac{\partial^2 \cdot}{\partial y^2}$), should be close to zero in regions that vary smoothly and have a larger magnitude in areas where the gradient ($\mathbf{v}$) changes quickly.

To find f_{in} for discrete pixels, a finite difference discretization can be used. Let p represent a pixel in h and let $q \in N_p$ represent the four directly adjacent neighbours of p. The solution should satisfy Eqn. 6.5 (or Eqn. 6.6 if any neighbouring pixels q overlap with the patch boundary ∂h) [Pérez et al., 2003].

$$|N_p|\, f_{in_p} - \sum_{q \in N_p} f_{in_q} = \sum_{q \in N_p} \text{v}_{pq} \tag{6.5}$$

$$\sum_{q \in N_p \cap h} \left(f_{in_p} - f_{in_q}\right) = \sum_{q \in N_p \cap \partial h} f_{out_q} + \sum_{q \in N_p} \text{v}_{pq} \tag{6.6}$$

In our case the image gradient comes from finite differences in the source image x_j, i.e. $\text{v}_{pq} = x_{j_p} - x_{j_q}$. Meanwhile, the boundary values, f_{out_q}, come directly from the destination image, x_{i_q}. This system can be solved for all $p \in h$ using an iterative solver. In some cases, this interpolation can cause a smearing effect. For example, when there is a large difference between the boundary values at opposite ends of the patch, but the gradient within the patch (from x_j) is very low. To compensate for this, Perez et al. suggest using the original gradient (from x_i) if it is larger than the gradient from x_j (Eqn. 6.7) [Pérez et al., 2003]. We modify this to introduce the interpolation factor, α, that FPI uses to control the contribution of x_j to the convex combination. In this case, α controls which image gradients take precedence (Eqn. 6.8). This creates more variety in training samples, i.e. more ways in which two patches can be combined. It also helps create a self-supervised task with varying degrees of difficulty, ranging from very subtle to more prominent structural differences. Figure 6.1 demonstrates that this

formulation can blend patches seamlessly.

$$\mathrm{v}_{pq} = \begin{cases} x_{i_p} - x_{i_q} \text{ if } \left|x_{i_p} - x_{i_q}\right| > \left|x_{j_p} - x_{j_q}\right| \\ x_{j_p} - x_{j_q} \text{ otherwise} \end{cases} \tag{6.7}$$

$$\mathrm{v}_{pq} = \begin{cases} (1-\alpha)\left(x_{i_p} - x_{i_q}\right) & \text{if } \left|(1-\alpha)\left(x_{i_p} - x_{i_q}\right)\right| > \left|\alpha\left(x_{j_p} - x_{j_q}\right)\right| \\ \alpha\left(x_{j_p} - x_{j_q}\right) & \text{otherwise} \end{cases} \tag{6.8}$$

PII uses the same loss as FPI, which is essentially a pixel-wise regression of the interpolation factor α [Tan et al., 2022]. The loss is given in Eqn. 6.9:

$$\mathcal{L}_{\text{bce}} = -\widetilde{y}_{i_p} log A_s(\widetilde{x}_{i_p}) - (1 - \widetilde{y}_{i_p}) \log(1 - A_s(\widetilde{x}_{i_p})) \tag{6.9}$$

The inputs, $\widetilde{x}_i$, are training samples that contain a random patch with values $f_i n$, computed via Poisson blending as described above. The output of the model is used directly as an anomaly score, A_s. A diagram of the setup is given in Figure 6.2.

Architecture and Specifications: We follow the same network architecture as FPI, a wide

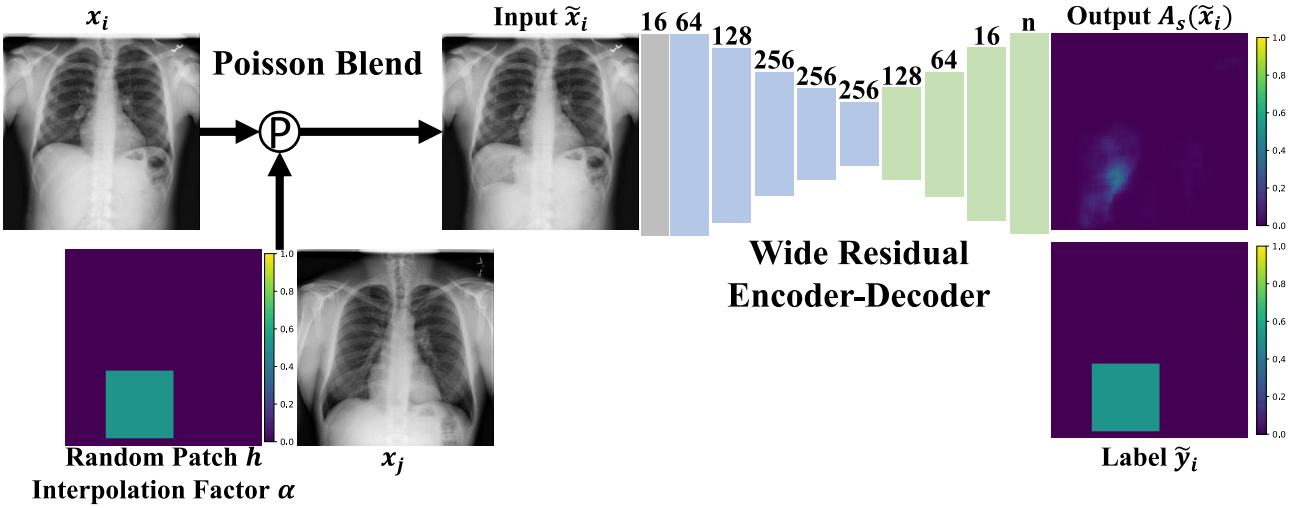


Figure 6.2: *Illustration of PII self-supervised training. The network architecture starts with a single convolutional layer (gray), followed by residual blocks (blue/green). Values above each block indicate the number of feature channels in the convolutional layers. In all experiments we use a single output channel, i.e.* $n = 1$.

residual encoder-decoder [Tan et al., 2022]. The encoder is a standard wide residual network [Zagoruyko and Komodakis, 2016] with a width of 4 and depth of 14. The decoder has the same structure but in reverse. The output has the same shape as the input and uses a

sigmoid activation. Training is done using Adam ([Kingma and Ba, 2015]) with a learning rate of 10^{-3} for 50 epochs. The self-supervised task uses patches, h, that are randomly generated with size $h_s \sim U(0.1N, 0.4N)$, and center coordinates $h_c \sim U_2(0.1N, 0.9N)$. Each patch is also given a random interpolation factor $\alpha \sim U(0.05, 0.95)$.

Implementation: We use TensorFlowV1.15 and train on a Nvidia TITAN Xp GPU. Training on our largest dataset for 50 epochs takes about 11 hours. PII solves partial differential equations on the fly to generate training samples dynamically. To achieve this we use multiprocessing [McKerns et al., 2012] to generate samples in parallel. The code is available at https://github.com/jemtan/PII.

6.1.4 Evaluation and Results

To evaluate the performance of PII, we compare with an embedding-based method, a reconstruction-based method, and a self-supervised method. For the embedding-based method, we use Deep SVDD [Ruff et al., 2018] with a 6 layer convolutional neural network. Meanwhile, a vector-quantized variational autoencoder (VQ-VAE2) [Razavi et al., 2019a] is used as a reconstruction-based method. The VQ-VAE2 is trained using the same wide residual encoder-decoder architecture as PII, except the decoder is given the same capacity as the encoder to help produce better reconstructions. FPI [Tan et al., 2022], which our method builds upon, is used as a self-supervised benchmark method. To compare each method, we calculate the average precision (AP) for each of the datasets described below. Average precision is a scalar metric for the area under the precision-recall curve.

Data: Our first dataset is ChestX-ray14 [Wang et al., 2017], a public chest X-ray dataset with 108,948 images from 32,717 patients showing 14 pathological classes as well as a normal class. From this large dataset, we extract 43,322 posteroanterior (PA) views of adult patients (over 18) and split them into male (♂) and female (♀) partitions. All X-ray images are resized to 256x256 (down from 1024x1024) and normalized to have zero mean and unit standard deviation. The training/test split is summarized in Table 6.1.

The second dataset consists of a total of 13380 frames from 108 patients acquired during routine fetal ultrasound screening. This is an application where automated anomaly detection in screening services would be of most use. We use cardiac standard view planes [National Health Service and Public Health England, 2018], specifically 4-chamber heart (4CH) and 3-vessel and trachea (3VT) views, from a private and de-identified dataset of ultrasound videos. For each selected standard plane, 20 consecutive frames (10 before and 9 after) are extracted from the ultrasound videos. Images are 224x288 and are normalized to zero mean, unit standard deviation. Normal samples consist of healthy images from a single view (4CH/3VT) and anomalous images are composed of alternate views (3VT/4CH) as well as pathological hearts of the same view (4CH/3VT). For pathology we use cases of hypoplastic left heart syndrome (HLHS), a condition that affects the development of the left side of the heart [Simpson, 2000]. The training/test split is outlined in Table 6.1. The scans are of volunteers at 18-24 weeks gestation (Ethics: anonymous during review), in a fetal cardiology clinic, where patients are referred to from primary screening and secondary care sites. Video clips have been acquired on Toshiba Aplio i700, i800 and Philips EPIQ V7 G devices.

Results: We compare our method with recent state-of-the-art anomaly detection methods in Table 6.1.

Dataset	Chest X-ray		Fetal US	
	♂PA	**♀PA**	**4CH**	**3VT**
	Number of Images			
Normal Train	17852	14720	283×20	225×20
Normal Test	2634	2002	34×20	35×20
Anomalous Test	3366	2748	54×20	38×20
	Average Precision			
Deep SVDD	0.565	0.556	0.685	0.893
VQ-VAE2	0.503	0.516	0.617	0.578
FPI	0.533	0.586	0.658	0.710
PII	**0.690**	**0.703**	**0.723**	**0.929**

Table 6.1: Each dataset is presented in one column. The train-test split is shown for each partition (top). Note that ultrasound images are extracted from videos as 20 frame clips. Average precision is also listed for each method (bottom).

Example test images are shown in Figure 6.3. The VQ-VAE2 reconstruction error indicates that sharp edges are difficult to reproduce accurately. Meanwhile FPI is sensitive to sharp edges because of the patch artifacts produced during training. In contrast, PII is sensitive to specific areas that appear unusual.

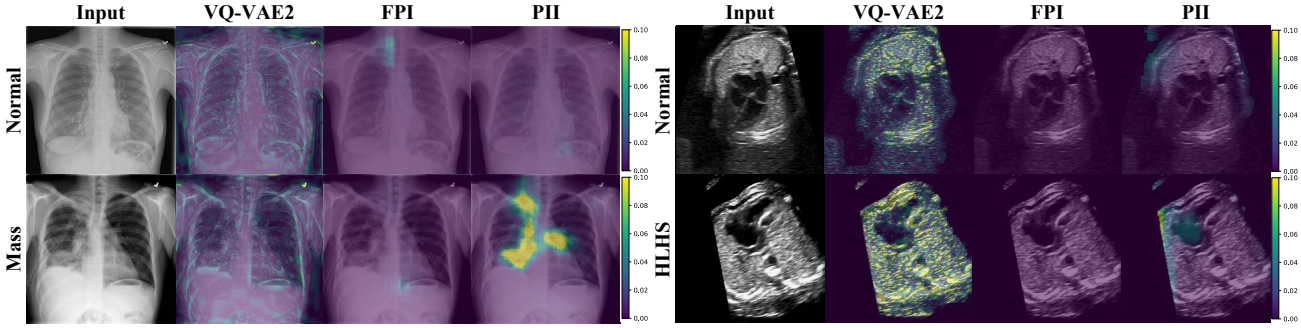
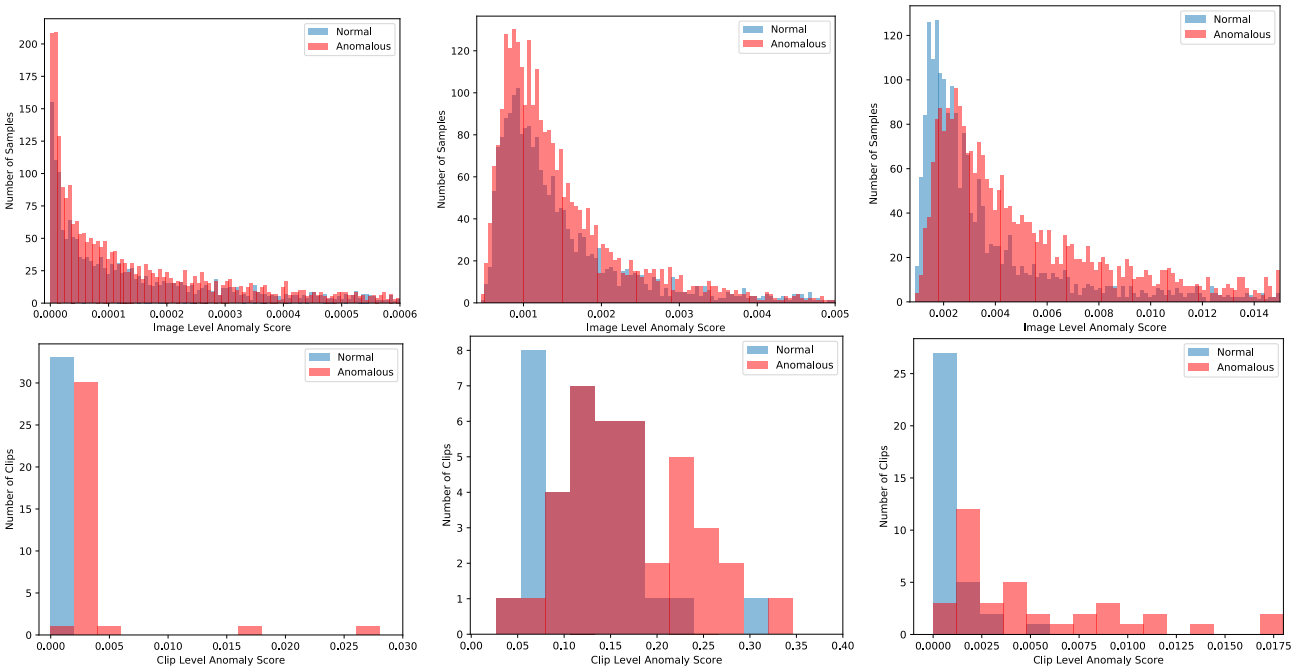


Figure 6.3: *Examples of test X-ray (left) and ultrasound (right) images with pixel-wise anomaly scores from each method. Note that the VQ-VAE2 reconstruction error is scaled down by a factor of 10.*



(a) *Deep SVDD* (b) *FPI* (c) *PII*

Figure 6.4: *Histograms of image level anomaly scores for Chest X-ray Female PA data (top) and clip level anomaly scores for Fetal US 3VT data (bottom).*

To see each method's ability to separate normal from anomalous, we plot histograms of anomaly scores for each method in Figure 6.4. The difficulty of these datasets is reflected in the fact that existing methods have almost no discriminative ability. On average, PII gives anomalous samples slightly higher scores than normal samples. The unusually high performance in the

3VT dataset is partly due to the small size of the dataset as seen in Figure 6.4.

Discussion: We have shown that our method is suitable to detect pathologies when they are considered anomalies compared to a training set that contains only healthy subjects. Training from only normal data is an important aspect in our field since a) data from healthy volunteers is usually available more easily, b) prevalence for certain conditions is low, thus collecting a well balanced training set is challenging and c) supervised methods would require in the ideal case equally many samples from every possible disease they are meant to detect. The latter is particularly a problem in rare diseases where the number of patients are very low in the global population.

6.1.5 Summary

In this work we have discussed an alternative to reconstruction-based anomaly detection methods. We base our method on the recently introduced FPI method, which formulates a self-supervised task through patch-interpolation based on normal data only. We advance this idea by introducing Poisson Image Interpolation, which mitigates interpolation issues for challenging data like chest X-Rays and fetal ultrasound examinations of the cardio-vascular system. In future work we will explore spatio-temporal support for PII, which is in particular relevant for ultrasound imaging.

6.2 MetaDetector: Detecting Outliers by Learning to Learn from Self-supervision

6.2.1 Overview

Using self-supervision in anomaly detection can increase sensitivity to subtle irregularities. However, increasing sensitivity to certain classes of outliers could result in decreased sensitivity to other types. While a single model may have limited coverage, an adaptive method could help

detect a broader range of outliers. Our proposed method explores whether meta learning can increase the adaptability of self-supervised methods. Meta learning is often employed in few-shot settings with labelled examples. To use it for anomaly detection, where labelled support data is usually not available, we instead construct a self-supervised task using the test input itself and reference samples from the normal training data. Specifically, patches from the test image are introduced into normal reference images. This forms the basis of the few-shot task. During training, the same few-shot process is used, but the test/query image is substituted with a normal training image that contains a synthetic irregularity. Meta learning is then used to learn how to learn from the few-shot task by computing second order gradients. Given the importance of screening applications, *e.g.*, in healthcare or security, any adaptability in the method must be counterbalanced with robustness. Thus, we add strong regularization by i) restricting meta learning to only layers near the bottleneck of our encoder-decoder architecture and ii) computing the loss at multiple points during the few-shot process.

6.2.2 Introduction

The main goal in outlier detection is to detect unanticipated irregularities. Many important problems can be framed in this way, including content moderation, security, and disease screening. All of these tasks can be very taxing on workers and phenomena such as inattentional blindness can make it particularly difficult to detect unexpected stimuli [Drew et al., 2013]. Machine learning methods have the potential to assist in many of these tasks. While most methods require labelled data to achieve expert-level performance, *e.g.*, supervised methods for detecting breast cancer [Wu et al., 2019] or retinal disease [De Fauw et al., 2018], self-supervised methods have recently begun to close the gap [Oord et al., 2018, Hénaff et al., 2019, Chen et al., 2020b, He et al., 2020].

Similar self-supervised approaches also exist in outlier detection [Golan and El-Yaniv, 2018, Tack et al., 2020]. However, applications such as medical imaging can require very specific and subtle features that are difficult to learn in an unsupervised manner. Many diseases can only be detected by those with domain expertise; but being a specialist does not always improve

detection. The "cost of expertise" is a bias toward familiar patterns, and can cause rigidity in perception of new stimuli. For example, one study found that physicians have a tendency to make diagnoses related to their speciality, even when examining cases outside of their domain [Hashem et al., 2003]. This is especially problematic in open-ended problems such as outlier detection because there are no restrictions on what pathologies may appear.

As such, anomalous features can be subtle and disease specific, but can also vary immensely across pathologies. Recent methods have increased sensitivity to subtle outliers through self-supervision [Tan et al., 2022, Tan et al., 2021a]. Our aim in this work is to explore whether meta learning can improve the adaptability of these methods. We construct a self-supervised few-shot task to be used during both training and testing. During training, we use second order gradients [Finn et al., 2017] to optimize for initial parameters that are adaptable to different tasks. In testing, the few-shot task gives the model a chance to adapt to features in the test data, potentially priming the model for better detection.

6.2.3 Related Work

Many methods have been developed to tackle outlier detection from different perspectives. Reconstruction-based methods use auto-encoders [Baur et al., 2021] or generative models [Schlegl et al., 2019] to reproduce or restore [Marimont and Tarroni, 2021] the normal components of the image. Errors in the reconstruction are used to highlight abnormalities. This is most effective for abnormalities that exhibit large intensity differences. Self-supervised methods are trained on proxy tasks that can either exploit (i) whole image augmentations that help the network to learn holistic features and major landmarks in the normal data [Golan and El-Yaniv, 2018, Tack et al., 2020] or (ii) patch-based augmentations that increase sensitivity to sub-image anomalies [Tan et al., 2022, Tan et al., 2021a, Li et al., 2021]. The design of the self-supervised task can influence which types of features are learned and consequently which types of anomalies are detected. This can help increase sensitivity to specific types of irregularities, but it can also limit detection of other types of anomalies.

Meta learning can be applied to few-shot problems to allow models to adapt quickly to new

tasks. One of the most ubiquitous strategies in this area is model-agnostic meta-learning (MAML) [Finn et al., 2017]. While first order gradients point toward parameters that give better outputs, second order gradients point toward parameters that give better few-shot gradients. By backpropagating through the few-shot optimization steps, MAML aims to learn an initialization point that benefits the most from the few-shot gradients. This strategy is typically used in settings with labelled examples, but there are also extensions in unsupervised settings that use clustering [Hsu et al., 2019]. There are even few-shot methods in outlier detection; however, these exploit a small amount of real anomalous data [Ding et al., 2021], or use multi-class data organized into normal and anomalous categories [Jeong and Kim, 2020]. The goal of this work is to explore meta learning in a setting where the few-shot task is self-supervised.

6.2.4 Method

Our proposed method, MetaDetector, is a meta learning approach for self-supervised outlier detection. In this section we briefly describe the self-supervised tasks, the meta learning process, and the regularization involved in training.

The two self-supervised tasks used in this method are foreign patch interpolation (FPI) [Tan et al., 2022] and Poisson image interpolation (PII) [Tan et al., 2021a]. In both tasks, a random patch is taken from one image and introduced into another image. This is done through linear interpolation in the case of FPI and by Poisson image editing [Pérez et al., 2003] in the case of PII. In both cases, the corresponding label is a mask of the altered patch that is scaled by the blending factor. Figure 6.5 depicts an example of FPI, where a patch in image x_i has been altered to produce $\widetilde{x}_i$. Using $\widetilde{x}_i$ as an input, the output of the network, $A_s(\widetilde{x}_i)$, is compared with the label $\widetilde{y}_i$ using a binary cross-entropy loss.

The meta learning component of our method is based on MAML [Finn et al., 2017] and it involves an inner and outer optimization loop. The inner loop is the few-shot optimization process. It involves a query image x_q, which may or may not be anomalous, and a reference image x_r which is a normal image from the training data. Random patches from the query image are introduced into the reference image using FPI, producing $\widetilde{x}_r$. This creates an input

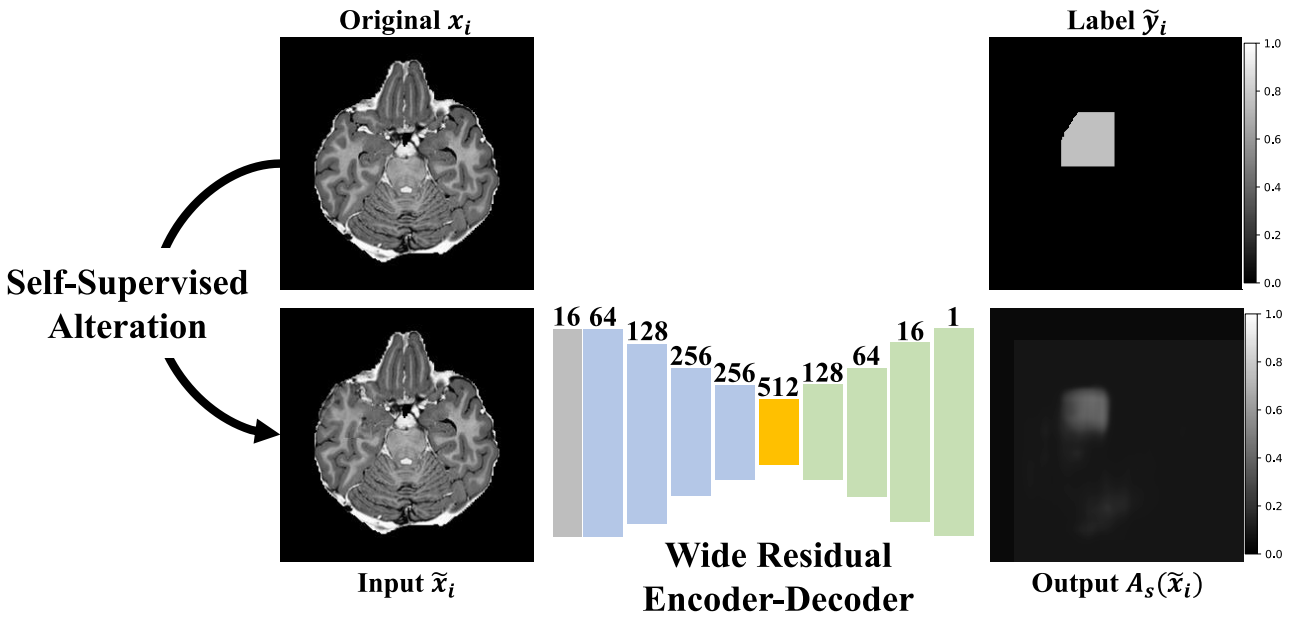


Figure 6.5: *Network architecture with example inputs, labels, and outputs. Quantities above each residual block indicate the number of feature channels.*

and label, as shown in Figure 6.5, which can be used to take one optimization step in the inner loop. This can be repeated k times depending on how many steps are desired in the few-shot process. In each step, the query image remains the same, but a new reference image and a new random patch are selected. In all of our experiments we use $k = 2$. This few-shot process is applied during testing and training. In testing, the query image is an unknown sample, x_q, but during training the query image is a normal sample (from the training data) that has been altered with FPI or PII, *i.e.*, $\widetilde{x}_q$.

The loss for both self-supervised tasks is a pixel-wise regression using binary cross-entropy, as defined by Eqn. 6.10. In this equation, f_θ is used to represent the model parameterized by θ. The parameter updates in the inner loop are characterized by Eqn. 6.11. This is a standard gradient update. Meanwhile Eqn. 6.12 specifies the parameter updates for the outer loop. Note that the loss is evaluated using the updated parameters from the inner loop, ϕ, and the query sample, $\widetilde{x}_q$. However, the gradient is taken with respect to the initial parameters, θ, which means that gradients must flow through the gradient steps of the inner loop.

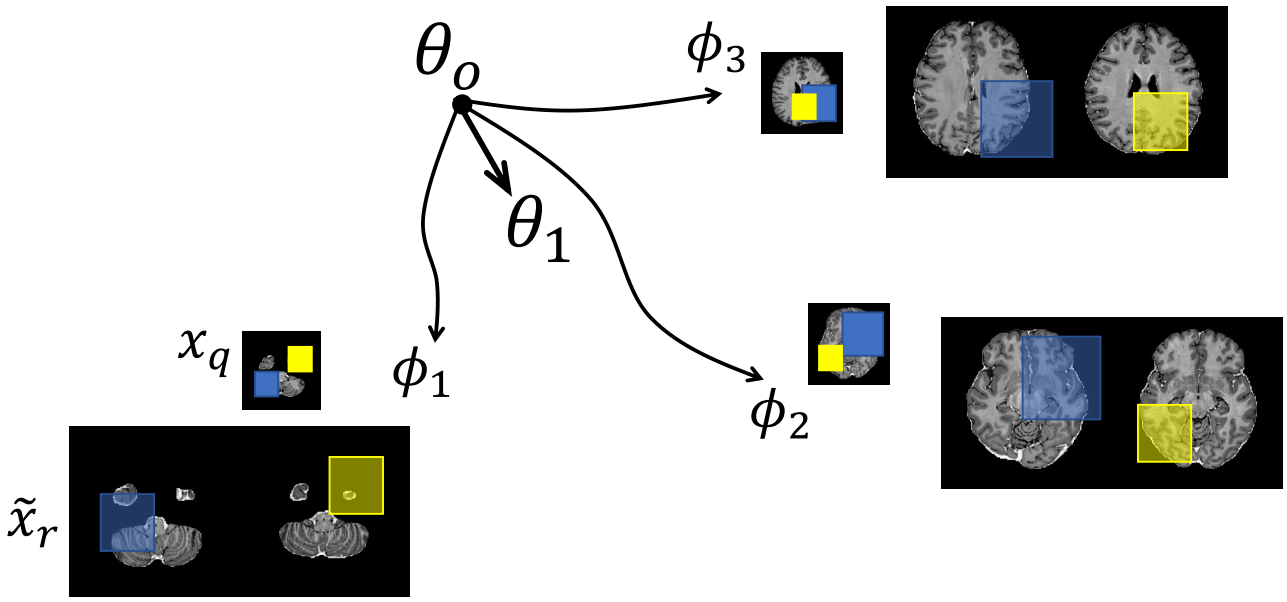


Figure 6.6: *Meta learning process with self-supervised few-shot task. Random patches, highlighted in blue and yellow, are taken from each query image x_q and introduced into reference images, x_r, forming $\widetilde{x}_r$. Each ϕ is the result of few-shot optimization using these synthetically altered samples, $\widetilde{x}_r$. Second order gradients from each few-shot task are aggregated to improve the initialization parameters, θ.*

$$\mathcal{L}_{\text{bce}}(\widetilde{x}_r, \widetilde{y}_r, f_\theta) = -\widetilde{y}_r log f_\theta(\widetilde{x}_r) - (1 - \widetilde{y}_r)\log(1 - f_\theta(\widetilde{x}_r)) \tag{6.10}$$

$$\phi_i = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{bce}}(\widetilde{x}_r, \widetilde{y}_r, f_\theta) \tag{6.11}$$

$$\theta_i = \theta - \beta \nabla_\theta \mathcal{L}_{\text{bce}}(\widetilde{x}_q, \widetilde{y}_q, f_\phi) \tag{6.12}$$

Note that Eqn. 6.11 and Eqn. 6.12 are simplified for legibility. In reality, ϕ_i is the result of several gradient steps, and θ_i is updated using an aggregate of multiple inner loops. This is depicted in Figure 6.6. In our experiments, the step sizes α and β are set to 1e$-$2 and 1e$-$3 respectively.

The last component of our method is the regularization. Since the model is being trained for fast adaptation, the predictions from the model could change drastically after only a few gradient steps. This is useful for adaptability, but it could also negatively impact robustness. We add strong regularization to the optimization process in two ways. First, we restrict meta learning to parameters in the bottleneck residual block (orange in Figure 6.5). The rest of the parameters are learned using standard self-supervised training (as in FPI [Tan et al., 2022] and

PII [Tan et al., 2021a]) using only the query samples, $\widetilde{x}_q$. This limits the number of parameters that can change during few-shot adaptation and drastically reduces computational costs.

Our second form of regularization is a multi-step loss based on a strategy proposed in MAML++ [Antoniou et al., 2018]. The original MAML method computes outer loop gradients according to Eqn. 6.12, specifically using f_ϕ, the parameters that are reached after taking the final few-shot step. With a multi-step approach, we compute the loss after every few-shot step [Antoniou et al., 2018]. The total loss is a weighted sum, where coefficients increase linearly with step number.

To run MetaDetector in evaluation mode, only the few-shot process is performed. Patches from the test image, x_q, are introduced into normal reference images, x_r, which creates self-supervised samples $\widetilde{x}_r$. The model is trained on these samples to reach the adapted parameters, ϕ. The updated model is then used to run inference on the original test image, x_q. The output from this inference is used directly as an anomaly score map, $A_s(x_q)$.

6.2.5 Evaluation and Results

We train MetaDetector on normal brain MRI and abdominal CT data from the medical out-of-distribution (MOOD) analysis challenge [Zimmerer et al., 2020b]. Since this dataset only includes normal samples, we evaluate on a synthetic test dataset that includes spheres with uniform intensity shifts, noise additions, sink/source deformations, uniform translations, and reflections across axes of symmetry. The details and the code to reproduce these test cases are provided in Tan et. al [Tan et al., 2022].

Table 6.2: Average precision for brain and abdominal synthetic test data [Tan et al., 2022] (originally from the MOOD challenge [Zimmerer et al., 2020b]).

Anatomy	Method	Subject-level AP	Pixel-level AP
Brain	FPI [Tan et al., 2022]	0.9723	0.7319
	MetaDetector	**0.9989**	**0.8551**
Abdomen	FPI [Tan et al., 2022]	0.8854	**0.6229**
	MetaDetector	**0.9694**	0.1657

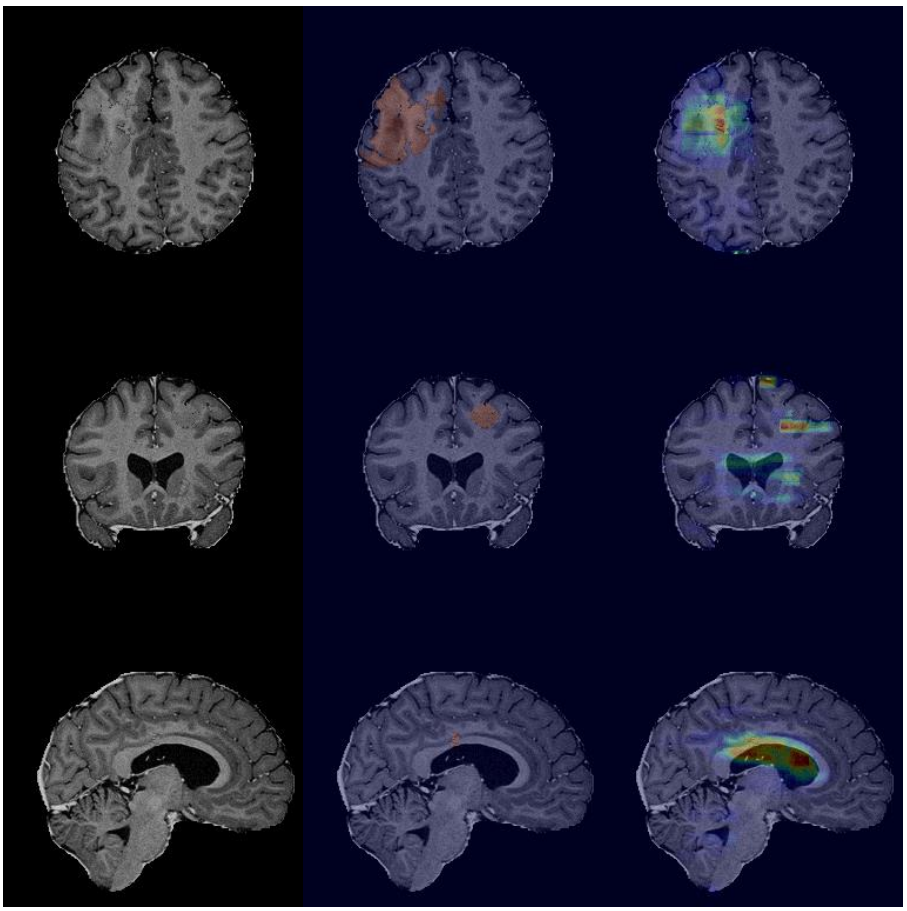


Figure 6.7: *Qualitative samples from the MOOD challenge [Zimmerer et al., 2020b]. Input, ground truth, and prediction (from left to right).*

Preliminary results are given in Table 6.2 and Figure 6.7. Overall, MetaDetector outperforms FPI [Tan et al., 2022] on synthetic test data. The low abdominal pixel-level score is likely due to a bias in the synthetic test data toward visibility in the coronal view. While FPI[Tan et al., 2022] was trained using 2D coronal slices [Tan et al., 2022], MetaDetector uses transverse slices at half the resolution (256x256) in order to meet computational restrictions. Qualitative examples in Figure 6.7 (provided by the MOOD organizers [Zimmerer et al., 2020b]) indicate that MetaDetector can localize different types of subtle abnormalities. However, false positives appear to be quite common. This could be either due to (i) synthetic training samples that are too subtle or (ii) the few-shot process priming the model to recognize healthy tissue from the query image as abnormal.

6.2.6 Discussion and Summary

We explore the use of meta learning in self-supervised outlier detection. Meta learning techniques such as MAML [Finn et al., 2017] can help models to quickly adapt to new data. We present an approach that uses self-supervision for the few-shot task, which allows training with only normal data. Early results indicate that the proposed method can outperform other self-supervised methods, such as FPI [Tan et al., 2022], on synthetic test data. This basic framework also has the potential to make outlier detection more adaptive and to better handle new stimuli. In future work, we aim to design more varied self-supervised tasks to encourage the model to rely more on information gained in the few-shot learning process.

Chapter 7

Conclusion

Outlier detection presents an interesting problem that is still challenging for current approaches. Much of the success in machine learning is built on the premise that large amounts of labelled examples are accessible before evaluation. But in the case of outlier detection, samples of outliers are typically not available *a priori*. As such, alternative objective functions must be explored. This thesis explores objective functions used in semi-supervised, unsupervised, and self-supervised learning.

Semi-supervised methods can make use of unlabelled data, but they rely on supervised learning to guide the process. When the distribution of unlabelled data differs considerably from the distribution of labelled data, these methods can actually make performance worse (Chapter 3.1). This can limit applicability in settings such as fetal ultrasound, where data is unbalanced. Although some sources of unlabelled data may be evenly sorted into major classes, the video data acquired during fetal examinations is heavily skewed toward a background class. In fact, only a small percentage of frames contain sufficient diagnostic information to be of interest to sonographers. Even after these key frames are extracted, diagnosing pathology can be challenging and typically requires supervised learning (Chapter 3.2).

Unsupervised learning is another option when using unlabelled data. Some unsupervised methods are based on clustering or contrasting which use underlying assumptions of the data distribution and/or prior knowledge in the form of augmentations. These assumptions may not

hold for problems in outlier detection, where the training data only contains one class and the relevant features can change depending on the deviations in the test sample. Chapter 4 explores an approach using divergent search to learn a repertoire of different clustering behaviors. The appropriate sorting behavior can then be deployed during evaluation with a few-shot fine tuning process. This allows for unsupervised learning without relying on predefined biases. Naturally, methods that use an appropriate bias can achieve higher performance. But the results also show that divergent search does in fact find useful features. As such, it may act as an alternative strategy when appropriate biases are not known a priori.

Learning from the weaknesses of existing unsupervised methods, Chapter 5 proposes a self-supervised task designed specifically for outlier detection in medical imaging. The goal of foreign patch interpolation is to encourage the network to learn the characteristic features of normal data. This is done by training the network to identify synthetic deviations and to estimate the degree to which normal has been perturbed. This approach overcomes problems with image space distances, seen in reconstruction-based methods, while also being sensitive to more subtle irregularities that existing self-supervised methods can struggle with. As such, it can complement existing methods and improve performance for certain types of outliers which are particularly relevant for medical imaging.

Building on the method developed in Chapter 5, Chapter 6.1 uses Poisson image editing [Pérez et al., 2003] to improve the quality of the image alterations. This produces more natural looking irregularities and reduces overfitting to artifacts introduced during anomaly synthesis. Chapter 6.2 proposes a step further, taking advantage of the adaptive properties of some meta-learning methods. In this setup the self-supervised task is repurposed for few-shot learning. This gives the model a chance to adapt to the current data (e.g. a specific neighbourhood of slices) or potentially to irregular characteristics of the test sample.

7.1 Limitations

Every outlier detection approach subsumes some assumptions, either implicitly or explicitly. The proposed self-supervised methods are no exception. These assumptions can greatly impact a method's suitability for different types of data. For reconstruction-based methods, outlier scores are computed as differences in input space. Intuitively, these metrics are more meaningful in lower dimensional data or for gross abnormalities which are effectively lower dimensional, i.e. still visible at greatly reduced resolutions. In contrast, the proposed methods can be sensitive to specific features, based on their self-supervised tasks. For these methods it can be less obvious which types of outliers will be identified accurately. Since it is generally not possible to predict or control which types of outliers will appear, it is important to at least be aware of a method's blind spots.

Studying the blind spots of each method can also provide insights into how these self-supervised tasks can be improved. Poisson image editing produces superior blending of foreign patches, but it can sometimes create exceedingly natural abnormalities that approximate normality. On one hand, it is important to explore the boundary of normality, but on the other hand, these differences may have limited practical relevance. Another issue is that the patches are selected randomly, but Poisson image editing was originally designed for manually selected patches. Typically, the source patch contains a subject while the destination patch is a background texture. Improving the patch selection process could improve the quality of synthetic anomalies and their similarity to real anomalies. However, it is not obvious how to determine general patch selection rules for a given dataset. It would likely require specific information about the structures in the image (e.g. segmentation labels of important structures) and may introduce unwanted biases.

Another aspect that could be improved is increasing the diversity of synthetic anomalies. Using a variety of anomaly generation techniques could increase the generalization to more types of real anomalies. This could also make better use of the proposed meta-learning approach. The purpose of using meta-learning in this context is to find a general minimum that is near to many task/anomaly specific minima. Currently, the same method is used to synthesize all

anomalies, so the network may converge to a minimum that is only relevant for this style of synthetic anomalies.

7.2 Future Work

Although perfect outlier detection without assumptions is an impossible task [Zhang et al., 2021], there are still many settings that could benefit from imperfect, but practical solutions. There are several areas of future work that can help bring these methods closer to the goal of helping clinicians. Firstly, it is not always possible to predict or control which types of outliers will appear. As such, it would be helpful to have an automated way of characterizing which types of outliers will be detectable. This could be accomplished through comprehensive empirical evaluation or by more theoretical study of generalization bounds. It could also take the form of more precise definitions of the assumptions made by each method.

Performance itself could also be improved by synthesizing outliers of higher quality. This could mean i) synthesizing anomalies that target the structures of highest importance, ii) new ways of generating more relevant transformations/deformations, or iii) simply increasing the diversity of anomaly generation techniques. A signature component of the proposed methods is the reliance on intra-class differences as a source of irregularity (via patch blending). New ways of generating sources of irregularity may provide more relevant, realistic, or varied synthetic anomalies. It is important to remember that the main objective is to learn features that are present in the normal data. In other words, the synthetic anomalies only exist for the purpose of providing contrast. In future work, it may be helpful to perform a systematic study of different types of contrasting classes and determine whether certain combinations provide further benefit.

Using more complete models of normal data could also lead to better anomaly detection. Having a deeper understanding of the underlying structures and anatomy behind the observations in an image could improve detection of semantic defects. However, this would likely require substantial amounts of effort, either in collecting more complete data or in creating custom models based on prior knowledge. Efforts such as the UK Biobank [Sudlow et al., 2015], which

contains scans from more than 100 healthy subjects, could support the creation of atlases or even functional models of specific organs. Meanwhile, more advanced generative methods could also help by learning better internal representations, e.g. separating texture and pose [Karras et al., 2021].

Robust performance for clinical practice may require the use of different strategies in concert. Each approach has blind spots and hybrid methods may offer more comprehensive coverage. Even clinical experts condone second opinions, so it is only natural to provide several answers. All in all, outlier detection still has many remaining challenges and there is much to gain in solving them.

Bibliography

[Alaverdyan et al., 2020] Alaverdyan, Z., Jung, J., Bouet, R., and Lartizien, C. (2020). Regularized siamese neural network for unsupervised outlier detection on brain multiparametric magnetic resonance imaging: application to epilepsy lesion screening. *Medical image analysis*, 60:101618.

[Alex et al., 2017] Alex, V., KP, M. S., Chennamsetty, S. S., and Krishnamurthi, G. (2017). Generative adversarial networks for brain lesion detection. In *Medical Imaging 2017: Image Processing*, volume 10133, page 101330G. International Society for Optics and Photonics.

[Antoniou et al., 2018] Antoniou, A., Edwards, H., and Storkey, A. (2018). How to train your maml. In *International Conference on Learning Representations*.

[Arjovsky et al., 2017] Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

[Arnaout et al., 2018] Arnaout, R., Curran, L., Chinn, E., Zhao, Y., and Moon-Grady, A. (2018). Deep-learning models improve on community-level diagnosis for common congenital heart disease lesions. *arXiv preprint arXiv:1809.06993*.

[Athiwaratkun et al., 2019] Athiwaratkun, B., Finzi, M., Izmailov, P., and Wilson, A. G. (2019). There are many consistent explanations of unlabeled data: Why you should average. *ICLR*.

[Atlason et al., 2019] Atlason, H. E., Love, A., Sigurdsson, S., Gudnason, V., and Ellingsen, L. M. (2019). Unsupervised brain lesion segmentation from mri using a convolutional autoencoder. In *Medical Imaging 2019: Image Processing*, volume 10949, page 109491H. International Society for Optics and Photonics.

[Ballard, 1987] Ballard, D. H. (1987). Modular learning in neural networks. In *Aaai*, volume 647, pages 279–284.

[Baumgartner et al., 2017] Baumgartner, C. F., Kamnitsas, K., Matthew, J., Fletcher, T. P., Smith, S., Koch, L. M., Kainz, B., and Rueckert, D. (2017). Sononet: real-time detection and localisation of fetal standard scan planes in freehand ultrasound. *IEEE transactions on medical imaging*, 36(11):2204–2215.

[Baur et al., 2021] Baur, C., Denner, S., Wiestler, B., Navab, N., and Albarqouni, S. (2021). Autoencoders for unsupervised anomaly segmentation in brain mr images: a comparative study. *Medical Image Analysis*, 69:101952.

[Baur et al., 2020] Baur, C., Wiestler, B., Albarqouni, S., and Navab, N. (2020). Scale-space autoencoders for unsupervised anomaly segmentation in brain mri. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 552–561. Springer.

[Bellman, 2015] Bellman, R. E. (2015). *Adaptive control processes*. Princeton university press.

[Bengio et al., 2009] Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

[Bennasar et al., 2010] Bennasar, M., Martinez, J., Gomez, O., Bartrons, J., Olivella, A., Puerto, B., and Gratacós, E. (2010). Accuracy of four-dimensional spatiotemporal image correlation echocardiography in the prenatal diagnosis of congenital heart defects. *Ultrasound in Obstetrics and Gynecology*, 36(4):458–464.

[Berthelot et al., 2019] Berthelot, D., Raffel, C., Roy, A., and Goodfellow, I. (2019). Understanding and improving interpolation in autoencoders via an adversarial regularizer. *ICLR*.

[Bozorgtabar et al., 2020] Bozorgtabar, B., Mahapatra, D., Vray, G., and Thiran, J.-P. (2020). Salad: Self-supervised aggregation learning for anomaly detection on x-rays. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 468–478. Springer.

[Brant and Stanley, 2017] Brant, J. C. and Stanley, K. O. (2017). Minimal criterion coevolution: a new approach to open-ended search. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 67–74. ACM.

[Bruls and Kwee, 2020] Bruls, R. and Kwee, R. (2020). Workload for radiologists during on-call hours: dramatic increase in the past 15 years. *Insights into Imaging*, 11(1):1–7.

[Budd et al., 2021] Budd, S., Sinclair, M., Day, T., Vlontzos, A., Tan, J., Liu, T., Matthew, J., Skelton, E., Simpson, J., Razavi, R., et al. (2021). Detecting hypo-plastic left heart syndrome in fetal ultrasound via disease-specific atlas maps. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 207–217. Springer.

[Cai et al., 2018] Cai, Y., Sharma, H., Chatelain, P., and Noble, J. A. (2018). Multi-task sonoeyenet: Detection of fetal standardized planes assisted by generated sonographer attention maps. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 871–879.

[Caron et al., 2018] Caron, M., Bojanowski, P., Joulin, A., and Douze, M. (2018). Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149.

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

[Castro et al., 2019] Castro, D. C., Tan, J., Kainz, B., Konukoglu, E., and Glocker, B. (2019). Morpho-mnist: quantitative assessment and diagnostics for representation learning. *Journal of Machine Learning Research*, 20(178):1–29.

[Cauchy et al., 1847] Cauchy, A. et al. (1847). Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538.

[Cerrolaza et al., 2018] Cerrolaza, J. J., Li, Y., Biffi, C., Gomez, A., Sinclair, M., Matthew, J., Knight, C., Kainz, B., and Rueckert, D. (2018). 3d fetal skull reconstruction from 2dus via deep conditional generative networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 383–391.

[Chen et al., 2020a] Chen, C., Qin, C., Qiu, H., Ouyang, C., Wang, S., Chen, L., Tarroni, G., Bai, W., and Rueckert, D. (2020a). Realistic adversarial data augmentation for mr image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 667–677. Springer.

[Chen et al., 2015a] Chen, H., Ni, D., Qin, J., Li, S., Yang, X., Wang, T., and Heng, P. A. (2015a). Standard Plane Localization in Fetal Ultrasound via Domain Transferred Deep Neural Networks. *IEEE Journal of Biomedical and Health Informatics*, 19.

[Chen et al., 2015b] Chen, H., Ni, D., Qin, J., Li, S., Yang, X., Wang, T., and Heng, P. A. (2015b). Standard plane localization in fetal ultrasound via domain transferred deep neural networks. *IEEE journal of biomedical and health informatics*, 19(5):1627–1636.

[Chen et al., 2020b] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020b). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

[Chen and Konukoglu, 2018] Chen, X. and Konukoglu, E. (2018). Unsupervised detection of lesions in brain mri using constrained adversarial auto-encoders. In *MIDL Conference book*. MIDL.

[Cheplygina et al., 2019] Cheplygina, V., de Bruijne, M., and Pluim, J. P. (2019). Not-so-supervised: A survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical Image Analysis*, 54:280 – 296.

[Chong, 2020] Chong, K. F. E. (2020). A closer look at the approximation capabilities of neural networks. *ICLR*.

[Chotzoglou et al., 2021] Chotzoglou, E., Day, T., Tan, J., Matthew, J., Lloyd, D., Razavi, R., Simpson, J., Kainz, B., et al. (2021). Learning normal appearance for fetal anomaly screening: Application to the unsupervised detection of hypoplastic left heart syndrome. *Machine Learning for Biomedical Imaging*, 1(September 2021 issue):1–10.

[Cubuk et al., 2019] Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. (2019). Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123.

[Cubuk et al., 2020] Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. (2020). Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703.

[Cully et al., 2015] Cully, A., Clune, J., Tarapore, D., and Mouret, J.-B. (2015). Robots that can adapt like animals. *Nature*, 521(7553):503.

[Cybenko, 1989] Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.

[De Fauw et al., 2018] De Fauw, J., Ledsam, J. R., Romera-Paredes, B., Nikolov, S., Tomasev, N., Blackwell, S., Askham, H., Glorot, X., O'Donoghue, B., Visentin, D., et al. (2018). Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine*, 24(9):1342–1350.

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

[Ding et al., 2021] Ding, K., Zhou, Q., Tong, H., and Liu, H. (2021). Few-shot network anomaly detection via cross-network meta-learning. In *Proceedings of the Web Conference 2021*, pages 2448–2456.

[Donahue et al., 2017] Donahue, J., Krähenbühl, P., and Darrell, T. (2017). Adversarial feature learning. *ICLR*.

[Doncieux et al., 2019] Doncieux, S., Laflaquière, A., and Coninx, A. (2019). Novelty search: a theoretical perspective. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 99–106.

[Drew et al., 2013] Drew, T., Võ, M., and Wolfe, J. (2013). The invisible gorilla strikes again: sustained inattentional blindness in expert observers. *Psychological Science*, 24(9):1848–1853.

[Eaton-Rosen et al., 2018] Eaton-Rosen, Z., Bragman, F., Ourselin, S., and Cardoso, M. J. (2018). Improving data augmentation for medical image segmentation. *Medical Imaging with Deep Learning*.

[Finn et al., 2017] Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135.

[Folgoc et al., 2019] Folgoc, L. L., Castro, D. C., Tan, J., Khanal, B., Kamnitsas, K., Walker, I., Alansary, A., and Glocker, B. (2019). Controlling meshes via curvature: Spin transformations for pose-invariant shape processing. In *International Conference on Information Processing in Medical Imaging*, pages 221–234. Springer.

[Fukushima, 1980] Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202.

[Fukushima and Miyake, 1982] Fukushima, K. and Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*, pages 267–285. Springer.

[Funahashi, 1989] Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural networks*, 2(3):183–192.

[Gal and Ghahramani, 2016] Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059.

[Glorot and Bengio, 2010] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings.

[Golan and El-Yaniv, 2018] Golan, I. and El-Yaniv, R. (2018). Deep anomaly detection using geometric transformations. In *Advances in Neural Information Processing Systems*, pages 9758–9769.

[Goodfellow et al., 2014] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.

[Grandvalet and Bengio, 2005] Grandvalet, Y. and Bengio, Y. (2005). Semi-supervised learning by entropy minimization. *NeurIPS*.

[Gretton et al., 2007] Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., and Smola, A. (2007). A kernel method for the two-sample-problem. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press.

[Gulrajani et al., 2017] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

[Hadsell et al., 2006] Hadsell, R., Chopra, S., and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

[Han et al., 2018] Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pages 8527–8537.

[Hashem et al., 2003] Hashem, A., Chi, M. T., and Friedman, C. P. (2003). Medical errors as a result of specialization. *Journal of biomedical informatics*, 36(1-2):61–69.

[He et al., 2020] He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

[Hénaff et al., 2019] Hénaff, O. J., Srinivas, A., De Fauw, J., Razavi, A., Doersch, C., Eslami, S., and Oord, A. v. d. (2019). Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*.

[Hendrycks et al., 2019] Hendrycks, D., Mazeika, M., and Dietterich, T. (2019). Deep anomaly detection with outlier exposure. *International Conference on Learning Representations*.

[Hinton et al., 2015] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

[Hornik, 1991] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257.

[Hornik et al., 1989] Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

[Hsu et al., 2019] Hsu, K., Levine, S., and Finn, C. (2019). Unsupervised learning via meta-learning. *ICLR*.

[Huang and Belongie, 2017] Huang, X. and Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510.

[Iqbal, 2018] Iqbal, H. (2018). Harisiqbal88/plotneuralnet v1.0.0.

[Izmailov et al., 2018] Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. (2018). Averaging weights leads to wider optima and better generalization. *Uncertainty in Artificial Intelligence*.

[Jeong and Kim, 2020] Jeong, T. and Kim, H. (2020). Ood-maml: Meta-learning for few-shot out-of-distribution detection and classification. *Advances in Neural Information Processing Systems*, 33.

[Karras et al., 2018] Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2018). Progressive growing of gans for improved quality, stability, and variation. *ICLR*.

[Karras et al., 2021] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., and Aila, T. (2021). Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34.

[Karras et al., 2019] Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

[Karras et al., 2020] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119.

[Kendall et al., 2018] Kendall, A., Gal, Y., and Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.

[Khodadadeh et al., 2019] Khodadadeh, S., Boloni, L., and Shah, M. (2019). Unsupervised meta-learning for few-shot image classification. In *Advances in Neural Information Processing Systems*, pages 10132–10142.

[Kim et al., 2018] Kim, B., Kim, K. C., Park, Y., Kwon, J.-Y., Jang, J., and Seo, J. K. (2018). Machine-learning-based automatic identification of fetal abdominal circumference from ultrasound images. *Physiological measurement*, 39(10):105007.

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. *ICLR*.

[Kingma and Welling, 2014] Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. *International Conference on Learning Representations*.

[Kong et al., 2018a] Kong, P., Ni, D., Chen, S., Li, S., Wang, T., and Lei, B. (2018a). Automatic and efficient standard plane recognition in fetal ultrasound images via multi-scale dense networks. In *Data Driven Treatment Response Assessment and Preterm, Perinatal, and Paediatric Image Analysis*, pages 160–168. Springer.

[Kong et al., 2018b] Kong, P., Ni, D., Chen, S., Wang, T., and Lei, B. (2018b). Automatic and Efficient Standard Plane Recognition in Fetal Ultrasound Images via Multi-scale Dense Networks. *Data Driven Treatment Response Assessment and Preterm, Perinatal, and Paediatric Image Analysis, LNCS*, 11076.

[Krizhevsky, 2009] Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. Technical report, University of Toronto.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105.

[Laine and Aila, 2017] Laine, S. and Aila, T. (2017). Temporal ensembling for semi-supervised learning. *ICLR*.

[Lake et al., 2015] Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

[Lecun, 1987] Lecun, Y. (1987). *PhD thesis: Modeles connexionnistes de l'apprentissage (connectionist learning models)*. PhD thesis, Universite P. et M. Curie (Paris 6).

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

[Lehman et al., 2018] Lehman, J., Chen, J., Clune, J., and Stanley, K. O. (2018). Safe mutations for deep and recurrent neural networks through output gradients. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 117–124. ACM.

[Lehman and Stanley, 2011a] Lehman, J. and Stanley, K. O. (2011a). Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2):189–223.

[Lehman and Stanley, 2011b] Lehman, J. and Stanley, K. O. (2011b). Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 211–218.

[Leshno et al., 1993] Leshno, M., Lin, V. Y., Pinkus, A., and Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867.

[Li et al., 2021] Li, C.-L., Sohn, K., Yoon, J., and Pfister, T. (2021). Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9664–9674.

[Linnainmaa, 1970] Linnainmaa, S. (1970). The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors. *Master's Thesis (in Finnish), Univ. Helsinki*, pages 6–7.

[Liu et al., 2020] Liu, T., Meng, Q., Vlontzos, A., Tan, J., Rueckert, D., and Kainz, B. (2020). Ultrasound video summarization using deep reinforcement learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 483–492. Springer.

[Lloyd, 1982] Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137.

[Lukasik et al., 2020] Lukasik, M., Bhojanapalli, S., Menon, A., and Kumar, S. (2020). Does label smoothing mitigate label noise? In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6448–6458. PMLR.

[Maaten and Hinton, 2008] Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

[MacQueen et al., 1967] MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

[Madry et al., 2018] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

[Mao et al., 2020] Mao, Y., Xue, F.-F., Wang, R., Zhang, J., Zheng, W.-S., and Liu, H. (2020). Abnormality detection in chest x-ray images using uncertainty prediction autoencoders. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 529–538. Springer.

[Marimont and Tarroni, 2021] Marimont, S. N. and Tarroni, G. (2021). Anomaly detection through latent space restoration using vector quantized variational autoencoders. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1764–1767. IEEE.

[Masci et al., 2011] Masci, J., Meier, U., Cireşan, D., and Schmidhuber, J. (2011). Stacked convolutional auto-encoders for hierarchical feature extraction. In *International conference on artificial neural networks*, pages 52–59. Springer.

[Mazziotta et al., 2001] Mazziotta, J., Toga, A., Evans, A., Fox, P., Lancaster, J., Zilles, K., Woods, R., Paus, T., Simpson, G., Pike, B., et al. (2001). A probabilistic atlas and reference system for the human brain: International consortium for brain mapping (icbm). *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 356(1412):1293–1322.

[McClelland et al., 1986] McClelland, J. L., Rumelhart, D. E., Group, P. R., et al. (1986). *Parallel distributed processing*, volume 2. MIT press Cambridge, MA.

[McKerns et al., 2012] McKerns, M. M., Strand, L., Sullivan, T., Fang, A., and Aivazis, M. A. (2012). Building a framework for predictive science. *arXiv preprint arXiv:1202.1056*.

[Meng et al., 2018] Meng, Q., Baumgartner, C., Sinclair, M., Housden, J., Rajchl, M., Gomez, A., Hou, B., Toussaint, N., Zimmer, V., Tan, J., et al. (2018). Automatic shadow detection in 2d ultrasound images. In *Data Driven Treatment Response Assessment and Preterm, Perinatal, and Paediatric Image Analysis*, pages 66–75. Springer.

[Menze et al., 2014] Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. (2014). The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024.

[Miyato et al., 2018a] Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018a). Spectral normalization for generative adversarial networks. *ICLR*.

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

[Müller et al., 2019] Müller, R., Kornblith, S., and Hinton, G. E. (2019). When does label smoothing help? In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

[Nair and Hinton, 2010] Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *ICML*.

[Nalepa et al., 2019] Nalepa, J., Marcinkiewicz, M., and Kawulok, M. (2019). Data augmentation for brain-tumor segmentation: a review. *Frontiers in computational neuroscience*, 13:83.

[National Congenital Anomaly and Rare Disease Registration Service, 2017] National Congenital Anomaly and Rare Disease Registration Service (2017). Congenital anomaly statistics 2017. Technical report, Public Health England.

[National Congenital Anomaly and Rare Disease Registration Service, 2019] National Congenital Anomaly and Rare Disease Registration Service (2019). Congenital anomaly statistics 2019. Technical report, Public Health England.

[National Congenital Anomaly and Rare Disease Registration Service, 2021] National Congenital Anomaly and Rare Disease Registration Service (2021). Guidance: Rare diseases. https://www.gov.uk/guidance/the-national-congenital-anomaly-and-rare-disease-registration-service-ncardrs. Accessed: 2022-03-22.

[National Health Service and Public Health England, 2018] National Health Service and Public Health England (2018). NHS Fetal Anomaly Screening Programme Handbook Valid from August 2018. Technical report, Public Health England.

[Nichol et al., 2018] Nichol, A., Achiam, J., and Schulman, J. (2018). On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*.

[Noroozi and Favaro, 2016] Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer.

[Oliver et al., 2018a] Oliver, A., Odena, A., Raffel, C., Cubuk, E. D., and Goodfellow, I. J. (2018a). Realistic evaluation of deep semi-supervised learning algorithms. *NeurIPS*.

[Oliver et al., 2018b] Oliver, A., Odena, A., Raffel, C. A., Cubuk, E. D., and Goodfellow, I. (2018b). Realistic evaluation of deep semi-supervised learning algorithms. *Advances in neural information processing systems*, 31.

[Oord et al., 2016] Oord, A. v. d., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., and Kavukcuoglu, K. (2016). Conditional image generation with pixelcnn decoders. *NeurIPS*.

[Oord et al., 2018] Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

[Pang et al., 2020] Pang, G., Shen, C., Cao, L., and van den Hengel, A. (2020). Deep learning for anomaly detection: A review.

[Pawlowski et al., 2018] Pawlowski, N., Lee, M. C., Rajchl, M., McDonagh, S., Ferrante, E., Kamnitsas, K., Cooke, S., Stevenson, S., Khetani, A., Newman, T., et al. (2018). Unsupervised lesion detection in brain ct using bayesian convolutional autoencoders. *MIDL*.

[Pérez et al., 2003] Pérez, P., Gangnet, M., and Blake, A. (2003). Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318.

[Petersen et al., 2021] Petersen, J., Köhler, G., Jäger, P., Full, P., Zimmerer, D., Maier-Hein, K., Roß, T., Adler, T., Reinke, A., and Maier-Hein, L. (2021). Medical Out-of-Distribution Analysis Challenge 2021.

[Pimentel et al., 2014] Pimentel, M. A., Clifton, D. A., Clifton, L., and Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, 99:215–249.

[Pinaya et al., 2021] Pinaya, W. H. L., Tudosiu, P.-D., Gray, R., Rees, G., Nachev, P., Ourselin, S., and Cardoso, M. J. (2021). Unsupervised brain anomaly detection and segmentation with transformers. *arXiv preprint arXiv:2102.11650*.

[Pinkus, 1999] Pinkus, A. (1999). Approximation theory of the mlp model in neural networks. *Acta numerica*, 8:143–195.

[Pinto et al., 2012] Pinto, N., Keenan, H., Minich, L., Puchalski, M., Heywood, M., and Botto, L. (2012). Barriers to prenatal detection of congenital heart disease: a population-based study. *Ultrasound in obstetrics & gynecology*, 40(4):418–425.

[Poličar et al., 2019] Poličar, P. G., Stražar, M., and Zupan, B. (2019). opentsne: a modular python library for t-sne dimensionality reduction and embedding. *bioRxiv*.

[Policar et al., 2019] Policar, P. G., Strazar, M., and Zupan, B. (2019). opentsne: a modular python library for t-sne dimensionality reduction and embedding. *BioRxiv*, page 731877.

[Pondenkandath et al., 2018] Pondenkandath, V., Alberti, M., Puran, S., Ingold, R., and Liwicki, M. (2018). Leveraging random label memorization for unsupervised pre-training. *Workshop on Integration of Deep Learning Theories at NeurIPS*.

[Pugh et al., 2016] Pugh, J. K., Soros, L. B., and Stanley, K. O. (2016). Quality diversity: A new frontier for evolutionary computation. *Frontiers in Robotics and AI*, 3:40.

[Radford et al., 2016] Radford, A., Metz, L., and Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. *ICLR*.

[Razavi et al., 2019a] Razavi, A., Oord, A. v. d., and Vinyals, O. (2019a). Generating diverse high-fidelity images with vq-vae-2. *ICLR Workshop DeepGenStruct*.

[Razavi et al., 2019b] Razavi, A., van den Oord, A., and Vinyals, O. (2019b). Generating diverse high-fidelity images with vq-vae-2. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

[Rezende et al., 2014] Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR.

[Ricklefs, 2010] Ricklefs, R. E. (2010). Evolutionary diversification, coevolution between populations and their antagonists, and the filling of niche space. *Proceedings of the National Academy of Sciences*, 107(4):1265–1272.

[Robbins and Monro, 1951] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407.

[Ruff et al., 2018] Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M. (2018). Deep one-class classification. In *International conference on machine learning*, pages 4393–4402.

[Rumelhart et al., 1985] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.

[Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

[Salimans et al., 2017] Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I. (2017). Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*.

[Schlegl et al., 2019] Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G., and Schmidt-Erfurth, U. (2019). f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54:30–44.

[Schlegl et al., 2017] Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G. (2017). Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pages 146–157. Springer.

[Schlüter et al., 2021] Schlüter, H. M., Tan, J., Hou, B., and Kainz, B. (2021). Self-supervised out-of-distribution detection and localization with natural synthetic anomalies (nsa). *arXiv preprint arXiv:2109.15222*.

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

[Simpson, 2000] Simpson, J. (2000). Hypoplastic left heart syndrome. *Ultrasound in obstetrics & gynecology: the official journal of the International Society of Ultrasound in Obstetrics and Gynecology*, 15(4):271–278.

[Sinclair et al., 2018] Sinclair, M., Baumgartner, C. F., Matthew, J., Bai, W., Martinez, J. C., Li, Y., Smith, S., Knight, C. L., Kainz, B., Hajnal, J., et al. (2018). Human-level performance on automatic head biometrics in fetal ultrasound using fully convolutional neural networks. In *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 714–717.

[Singh et al., 2008] Singh, A., Nowak, R., and Zhu, J. (2008). Unlabeled data: Now it helps, now it doesn't. *NeurIPS*.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

[Sudlow et al., 2015] Sudlow, C., Gallacher, J., Allen, N., Beral, V., Burton, P., Danesh, J., Downey, P., Elliott, P., Green, J., Landray, M., et al. (2015). Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine*, 12(3):e1001779.

[Szegedy et al., 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

[Szegedy et al., 2014] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. In *International Conference on Learning Representations*.

[Szerlip et al., 2015] Szerlip, P. A., Morse, G., Pugh, J. K., and Stanley, K. O. (2015). Unsupervised feature learning through divergent discriminative feature accumulation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[Tack et al., 2020] Tack, J., Mo, S., Jeong, J., and Shin, J. (2020). Csi: Novelty detection via contrastive learning on distributionally shifted instances. *arXiv preprint arXiv:2007.08176*.

[Tan et al., 2020] Tan, J., Au, A., Meng, Q., FinesilverSmith, S., Simpson, J., Rueckert, D., Razavi, R., Day, T., Lloyd, D., and Kainz, B. (2020). Automated detection of congenital heart disease in fetal ultrasound screening. In *Medical Ultrasound, and Preterm, Perinatal and Paediatric Image Analysis*, pages 243–252. Springer.

[Tan et al., 2019] Tan, J., Au, A., Meng, Q., and Kainz, B. (2019). Semi-supervised learning of fetal anatomy from ultrasound. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*, pages 157–164. Springer.

[Tan et al., 2022] Tan, J., Hou, B., Batten, J., Qiu, H., Kainz, B., et al. (2022). Detecting outliers with foreign patch interpolation. *Machine Learning for Biomedical Imaging*, 1(April 2022 issue):1–10.

[Tan et al., 2021a] Tan, J., Hou, B., Day, T., Simpson, J., Rueckert, D., and Kainz, B. (2021a). Detecting outliers with poisson image interpolation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 581–591. Springer.

[Tan and Kainz, 2020] Tan, J. and Kainz, B. (2020). Divergent search for image classification behaviors. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 91–92. https://doi.org/10.1145/3377929.3389973.

[Tan et al., 2021b] Tan, J., Kart, T., Hou, B., Batten, J., and Kainz, B. (2021b). Metadetector: Detecting outliers by learning to learn from self-supervision. In *Biomedical Image Registration, Domain Generalisation and Out-of-Distribution Analysis at MICCAI*, pages 119–126. Springer.

[Tang et al., 2020] Tang, Y.-X., Tang, Y.-B., Peng, Y., Yan, K., Bagheri, M., Redd, B. A., Brandon, C. J., Lu, Z., Han, M., Xiao, J., et al. (2020). Automated abnormality classification of chest radiographs using deep convolutional neural networks. *NPJ digital medicine*, 3(1):1–8.

[Tarvainen and Valpola, 2017] Tarvainen, A. and Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30.

[Tehrani et al., 2013] Tehrani, A. S. S., Lee, H., Mathews, S. C., Shore, A., Makary, M. A., Pronovost, P. J., and Newman-Toker, D. E. (2013). 25-year summary of us malpractice claims for diagnostic errors 1986–2010: an analysis from the national practitioner data bank. *BMJ quality & safety*, 22(8):672–680.

[Tuysuzoglu et al., 2018] Tuysuzoglu, A., Tan, J., Eissa, K., Kiraly, A. P., Diallo, M., and Kamen, A. (2018). Deep adversarial context-aware landmark detection for ultrasound imaging. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 151–158. Springer.

[Van den Oord et al., 2016] Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al. (2016). Conditional image generation with pixelcnn decoders. *Advances in neural information processing systems*, 29.

[Van Den Oord et al., 2017] Van Den Oord, A., Vinyals, O., et al. (2017). Neural discrete representation learning. *Advances in neural information processing systems*, 30.

[Van Oord et al., 2016] Van Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. (2016). Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR.

[Van Velzen et al., 2016] Van Velzen, C., Clur, S., Rijlaarsdam, M., Bax, C., Pajkrt, E., Heymans, M., Bekker, M., Hruda, J., de Groot, C., Blom, N., et al. (2016). Prenatal detection of congenital heart disease—results of a national screening programme. *BJOG: An International Journal of Obstetrics & Gynaecology*, 123(3):400–407.

[Verma et al., 2019] Verma, V., Lamb, A., Kannala, J., and Bengio, Y. (2019). Interpolation consistency training for semi-supervised learning. *IJCAI*.

[Vincent et al., 2010] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12).

[Vinyals et al., 2016] Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638.

[Wang et al., 2019] Wang, R., Lehman, J., Clune, J., and Stanley, K. O. (2019). Paired open-ended trailblazer (poet): Endlessly generating increasingly complex and diverse learning environments and their solutions. In *Proceedings of the Genetic and Evolutionary Computation Conference*, page 142–151. ACM.

[Wang et al., 2017] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., and Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106.

[Wei et al., 2018] Wei, Q., Ren, Y., Hou, R., Shi, B., Lo, J. Y., and Carin, L. (2018). Anomaly detection for medical images based on a one-class classification. In *Medical Imaging 2018: Computer-Aided Diagnosis*, volume 10575, page 105751M. International Society for Optics and Photonics.

[Werbos, 1982] Werbos, P. J. (1982). Applications of advances in nonlinear sensitivity analysis. In *System modeling and optimization*, pages 762–770. Springer.

[Werbos, 1994] Werbos, P. J. (1994). *The roots of backpropagation: from ordered derivatives to neural networks and political forecasting*, volume 1. John Wiley & Sons.

[World Health Organization et al., 2016] World Health Organization et al. (2016). *WHO recommendations on antenatal care for a positive pregnancy experience*. World Health Organization.

[Wu et al., 2019] Wu, N., Phang, J., Park, J., Shen, Y., Huang, Z., Zorin, M., Jastrzębski, S., Févry, T., Katsnelson, J., Kim, E., et al. (2019). Deep neural networks improve radiologists' performance in breast cancer screening. *IEEE transactions on medical imaging*, 39(4):1184–1194.

[Xia et al., 2020] Xia, T., Chartsias, A., and Tsaftaris, S. A. (2020). Pseudo-healthy synthesis with pathology disentanglement and adversarial learning. *Medical Image Analysis*, 64:101719.

[Xie et al., 2019] Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., and Le, Q. V. (2019). Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*.

[Xie et al., 2020] Xie, Q., Dai, Z., Hovy, E., Luong, T., and Le, Q. (2020). Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268.

[Yan et al., 2018a] Yan, K., Wang, X., Lu, L., and Summers, R. M. (2018a). Deeplesion: automated mining of large-scale lesion annotations and universal lesion detection with deep learning. *Journal of medical imaging*, 5(3):036501.

[Yan et al., 2018b] Yan, K., Wang, X., Lu, L., Zhang, L., Harrison, A. P., Bagheri, M., and Summers, R. M. (2018b). Deep lesion graphs in the wild: relationship learning and organization of signifi-

cant radiology image findings in a diverse large-scale lesion database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9261–9270.

[Yao et al., 2019] Yao, J., Pan, W., Ghosh, S., and Doshi-Velez, F. (2019). Quality of uncertainty quantification for bayesian neural network inference. *arXiv preprint arXiv:1906.09686*.

[Yeo et al., 2018] Yeo, L., Luewan, S., and Romero, R. (2018). Fetal intelligent navigation echocardiography (fine) detects 98% of congenital heart disease. *Journal of Ultrasound in Medicine*, 37(11):2577–2593.

[You et al., 2019] You, S., Tezcan, K. C., Chen, X., and Konukoglu, E. (2019). Unsupervised lesion detection via image restoration with a normative prior. In *International Conference on Medical Imaging with Deep Learning*, pages 540–556. PMLR.

[Yun et al., 2019] Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. (2019). Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032.

[Yuste, 2015] Yuste, R. (2015). From the neuron doctrine to neural networks. *Nature reviews neuroscience*, 16(8):487–497.

[Zagoruyko and Komodakis, 2016] Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. In Richard C. Wilson, E. R. H. and Smith, W. A. P., editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press.

[Zhang et al., 2017] Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2017). Understanding deep learning requires rethinking generalization. *ICLR*.

[Zhang et al., 2018] Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. (2018). mixup: Beyond empirical risk minimization. *International Conference on Learning Representations*.

[Zhang et al., 2021] Zhang, L., Goldstein, M., and Ranganath, R. (2021). Understanding failures in out-of-distribution detection with deep generative models. In *International Conference on Machine Learning*, pages 12427–12436. PMLR.

[Zhao et al., 2019] Zhao, S., Song, J., and Ermon, S. (2019). Infovae: Balancing learning and inference in variational autoencoders. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 5885–5892.

[Zimmerer et al., 2019a] Zimmerer, D., Isensee, F., Petersen, J., Kohl, S., and Maier-Hein, K. (2019a). Unsupervised anomaly localization using variational auto-encoders. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 289–297. Springer.

[Zimmerer et al., 2019b] Zimmerer, D., Kohl, S., Petersen, J., Isensee, F., and Maier-Hein, K. (2019b). Context-encoding variational autoencoder for unsupervised anomaly detection. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*.

[Zimmerer et al., 2020a] Zimmerer, D., Petersen, J., Köhler, G., Jäger, P., Full, P., Roß, T., Adler, T., Reinke, A., Maier-Hein, L., and Maier-Hein, K. (2020a). Medical out-of-distribution analysis challenge.

[Zimmerer et al., 2020b] Zimmerer, D., Petersen, J., Köhler, G., Jäger, P., Full, P., Roß, T., Adler, T., Reinke, A., Maier-Hein, L., and Maier-Hein, K. (2020b). Medical out-of-distribution analysis challenge.