Improving the Generalizability of Deep Learning Models for Polyp Segmentation by Optimizing for Consistency

Birk Sebastian Frostelid Torpmann-Hagen

Thesis submitted for the degree of
Master in Robotics and Intelligent Systems
60 credits

Department of Informatics
Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Spring 2022

Improving the Generalizability of Deep Learning Models for Polyp Segmentation by Optimizing for Consistency

Birk Sebastian Frostelid Torpmann-Hagen

Improving the Generalizability of Deep Learning Models for Polyp Segmentation by Optimizing for Consistency

http://www.duo.uio.no/

Printed: Reprosentralen, University of Oslo

Abstract

Contents

List of Figures

List of Tables

Preface

Chapter 1

Introduction

Colorectal cancer is one of the leading causes of cancer related deaths, causing approximately 900 thousand deaths worldwide per year (cite). Early detection thereof is as a consequence of significant importance. Polyps are often an early warning-sign of developing tumour, and early detection thereof can as a result significantly reduce fatality rates. Polyps are, however, often missed during colonoscopies, owing to the significant variability in the shapes and sizes of polyps, as well as the high degrees of similarity to surrounding tissue. Automatic segmentation of polyps via deep learning has the potential to significantly increase the likelihood of early detection and thus effective treatment.

Though there has been a wealth of work dedicated to developing such systems, with promising results, recent work in the field has highlighted that deep neural networks (DNNs) readily fail to maintain performance when deployed outside of lab-conditions. This is known as generalization failure, and has been shown to be ubiquitous across practically every application of deep learning. The deep learning community is still in the early stages of understanding exactly how and why such generalization failure is so ubiquitous. Consqeuently, developing methods and frameworks to combat generalisation remains an open problem.

This thesis attempts to address this problem by synthesizing and systematizing recent work in generalizability, generalizable methods, and recent attempts at inducing generalizability in polyp segmentation as presented in the EndoCV2021 challenge. A novel approach to increasing generalizability, based on recent work in the field, is also presented. The approach, named PLACEHOLDER ALGORITHM NAME, works by employing specific augmentation strategies to produce a so-called model of natural variation, intended to encapsulate the variability one might expect across different datasets and hence assist in inducing invariances in the model. This is acheived through the use of a specifically tailored loss, referred to as consistency loss, which punishes inconsistent predictions across the augmented and unaugmented folds irrespective of the correctness of the predictions. This endows the pipeline with the ability to more readily infer causally viable inductive biases by explicitly forcing the model to be robust to any combination of the aforementioned

transformations.

(...)

Chapter 2

Background

2.1 Colorectal Polyps, Medical Imaging, and Deep Learning

Polyps are small growths found in and around the inner lining of the large intestine. These polyps, also referred to as adenomas, can in time develop into cancerous tumours, or carcinomas, in a process known as the adenoma-carcinoma sequence [26]. Though the majority of polyps do not undergo this process, identifying polyps nonetheless constitutes an important step towards preventing colorectal cancer. Indeed, resection of these polyps has been shown to reduce the incidence of colorectal cancer by a significant margin [40].

Though colorectal cancer remains as one of the leading causes of cancer-related death worldwide (source), mortality rates have in recent years declined in large part to the increased use of screening colonoscopy and subsequent preemptive treatment. Polyps are by nature somewhat difficult to detect, however, and are routinely missed by clinicians, with miss rates reportedly ranging upwards of 27% for diminutive (<2.5mm) polyps [19, 29].

Reducing this miss rate and has the potential to further reduce the incidence of colorectal cancer. As a result, there has been a significant body of work dedicated to developing systems and techniques to aid in optimizing the screening procedure. One such example, referred to as chromoendoscopy, has been shown to reduce miss rates by (...) through the use of of specific dyes prior to the colonoscopy. Similarly, the use of narrow-band imaging techniques, wherein light of specific wavelengths specifically designed to highlight the textural differences between the polyps and the surrounding tissue, have been shown to reduce miss rates by (...).

These systems do, however, require specialized equipment, training and expertise to effectively employ. Thus, automatic polyp segmentation using deep learning and convolutional neural networks (CNNs) has also been identified as a possible diminutive detection method. This requires minimal training time on the part of the clinician, no additional equipment, and has been show to significantly increase detection rates when deployed in a clinical setting [5].

This has spurred on a large body of research dedicated to improving on the performance and expanding the capabilities of deep-learning based systems for polyp detection and segmentation. Several challenges been also held, namely the Endotect challenge [20], EndoCV2020 [2], EndoCV2021 [1], and more.

There are, however, still several hurdles to overcome; recent research has shown that even state of the art deep-learning pipelines are prone to generalization failure when deployed in practical settings, particularly when exposed to distributional shifts such as changes in demographics, imaging equipment, noise, and more despite exhibiting high performance on hold-out sets [6, 8, 14, 43]. This was further highlighted in the EndoCV2021 challenge, wherein submissions were evaluated on a hidden dataset collected from a different hospital than the training data. The results from this challenge demonstrated the pervasiveness of generalization failure, with every submitted model exhibiting significant performance reductions when evaluated on the hidden dataset [1].

Naturally, automatic segmentation systems are rendered practically useless should they fail to perform sufficiently outside the very carefully controlled conditions within which typical deep learning models are evaluated. Increasing the generalizability of the deep learning pipeline is as a result of significant interest.

2.2 Generalization failure in broader contexts

2.2.1 Generalization failure in Medical Imaging

Generalization failure is not, of course, unique to the polyp segmentation. Though medical imaging has in recent years proven to be one of the most promising applications of artificial intelligence and deep learning, having the capacity to significantly improve both the accuracy and efficiency of detection, diagnosis, and treatment of a wide variety of diseases [34], medical deep learning systems are nonetheless highly prone to generalization failure [8, 14]. In addition to the already limited capabilities of deep neural networks to generalize, medical domains are subject to a number of other exacerbating factors that make generalization all the more diffcult. Training data is often scarce, the pathologies that constitute the classification targets are unevenly distributed and often exhibit high degrees of within-class variability. Moreover, due to the sheer scope of the data involved, there are inevitably a significant number of confounding variables both during training and in deployment.

For instance, a deep-learning based classifier which successfully detected pneumonia in X-ray scans across a number of hospitals with striking accuracy was determined to be basing its predictions not on any lesions or otherwise pathologically relevant features in the images, but rather on a hospital-specific metal token that could be found in every image, which it used in conjunction with learning the pneumonia prevalence rate of for the respective hospitals. As a result, when deployed on data from hos-

pitals that it had not seen during training, the system failed to generalize [43]. In another study, it was shown that a classifier intended to detect diabetic retinopathy exhibited significant variability in performance depending on the type of camera used (cite). A similar study also showed that the same type of performance variability could be found when detecting skin-conditions across demographics with differing skin tones (cite). Finally, a model trained to detect and diagnose melanomas was shown to in large part be basing its predictions on whether or not it could detect any pre-surgical markings, used by doctors to assist in surgery preparation, in the vinicity of the lesion as opposed to actually learning anything about what the melanomas themselves[41].

2.2.2 Generalization Failure in General

Though generalization failure is perhaps best represented in medical domains, it can be shown that the phenomenon is pervasive in practically every application of deep learning. It has for instance been shown that CNNs trained on ImageNet, one of the largest and most diverse datasets in the domain of computer vision, are heavily biased towards textural features, and consequently fail when texture-altering style-transfer is applied, despite the shape and structure of the relevant object remaining recognizable[13]. Though this result is based on evaluation on synthetic data, it highlights a key property of deep learning pipelines: namely that they do not necessarily learn features that are causal - in other words, that they are intrinsic to the relevant object - inasmuch as they learn features that are highly correlated - in other words, features that are associated with the object but are not intrinsic to it. Though the texture of cat fur for instance is highly correlated with the "cat" class, it is not the fur that makes the cat. In Figure 2.1, for instance, it is clear that image (c) should be classified as a cat moreso than an elephant. Granted, this example is fairly synthetic, but a similar situation could arise if the classifier for instance was tested on a black-and-white image of a hairless cat.

(a) Texture image
81.4% **Indian elephant**
10.3% indri
8.2% black swan

(b) Content image
71.1% **tabby cat**
17.3% grey fox
3.3% Siamese cat

(c) Texture-shape cue conflict
63.9% **Indian elephant**
26.4% indri
9.6% black swan

Figure 2.1: Classifiers trained on ImageNet are biased towards textural features

Another characteristic of deep learning that supports this argument is the effectiveness of adversarial attacks [23], which specifically target

weaknesses in the inductive biases within DNNs through any number of means in an attempt to induce high rates of incorrect, yet highly confident predictions. Gradient-based adversarial attacks, for instance, use the gradients of the model to break even the most sophisticated and well-trained pipelines merely by adding some carefully crafted, yet visually imperceptible noise to the inputs [7]. Even without access to the gradients, there exists a multitude of so-called black-box attacks that only use output samples to generate similarly effective attacks (cite). Finally, it has been shown that adding minor visual distractions to objects, for example adding bits of tape or graffiti to stop signs, dramatically increases misclassification rates [11].

Even benign, but nonetheless confounding perturbations also have the potential to induce failure. It has for instance been shown that sophisticated natural language processing models can and readily do fail if one adds peripheral information to the input. (Example, citation)

2.3 Generalisability Theory

Exactly why and how DNNs seem to so persistently fail to generalize is a topic of ongoing research, and the available literature seems to suggest that the problem is multifaceted. This section is an attempt to summarize and distill the findings and analysis performed in the field. It will cover the theoretical basis of generalization and why one might expect DNNs to generalize, discuss the key characteristics of generalization failure and their origins, and finally introduce a probabilistic perspective of generalization.

2.3.1 Generalization through Empirical Risk Minimization

Naturally, deep learning would not have experienced as much of a revolution in the last decade or so if there was not some semblance of an expectation that their striking performance was generalisable and performant also outside the idealized settings typically involved in research. The theoretical basis that informs this belief in (most) modern deep learning pipelines is the idea of so-called empirical risk minimization, wherein it is assumed that the dataset upon which the model is trained is a representative sample of the distribution of all possible samples in the relevant domain. In other words, it assumes that the dataset is independently and identically distributed (iid) to the domain distribution. To better understand this assumption, it is beneficial to consider the it from first principles:

At the most fundamental level, the goal of machine learning is to learn a mapping between two spaces of objects X and Y. This mapping, namely the function $f : X \to Y$, maps some input object $x \in X$, an image for example, to a corresponding and application-relevant output object $y \in Y$, for instance a segmentation mask or a class probabilities. It is worth noting, however, that f is not as much a function in the mathematical sense as much as it is an abstraction of whatever ground-truth relationship that the deep learning system is intended to capture, and consequently cannot

typically be modelled explicitly. Instead, machine learning systems aim to find a representation of this mapping automatically by leveraging a training set $\{x_i, y_i\}_{0...n}$ to find a sufficiently performant approximation of f. This is referred to as supervised learning, and the resulting approximation found using the training set is denoted by $h : X \to \hat{Y}$, and typically referred to as a hypothesis.

To find such an approximation, we assume that there exists a joint probability distribution over X and Y, namely $P(x, y)$, and that the training data $\{x_i, y_i\}_{0...n}$ is drawn from this probability distribution such that the resulting sample distribution is independent and identically distributed to $P(x, y)$. This is the so-called iid assumption. By modelling the mapping as a joint probability distribution, one can model uncertainty in the predictions by expressing the output as a conditional probability $P(y|x)$. In conjunction with a loss-function $L(h(x), y)$ which measures the discrepancy between the hypothesis and the ground truth, these assumptions allows us to quantify the expected performance of a given hypothesis:

$$R(h) = \boldsymbol{E}[L(h(x), y)] = \int L(h(x), y) dP(x, y) \tag{2.1}$$

Using this framework, one can then find an iid-optimal hypothesis, often called a predictor, by finding the predictor h^* among a fixed class of functions (defined by network architecture) $\mathcal{H}$ that minimizes risk:

$$h^* = \arg\min_{h \in \mathcal{H}} R(h) \tag{2.2}$$

Since $P(x, y)$ is not known, however, one cannot compute $R(h)$ explicitly. Instead, the expected risk has to be estimated empirically, i.e by finding the arithmetic average of the risk associated with each prediction by the hypothesis over the training set:

$$R_{emp}(h) = \frac{1}{n} \sum_{i=1}^{n} L(h(x_i), y_i) \tag{2.3}$$

This risk can in turn be minimized with respect to the hypothesis class. This is called empirical risk minimization (ERM):

$$\hat{h} = \arg\min_{h \in \mathcal{H}} R_{emp}(h) \tag{2.4}$$

To reiterate, the central idea with this approach to machine learning is that the training data can be considered a finite iid sampling of the underlying distribution. As such, by the central limit theorem, the hold-out performance of the computed hypothesis will approach iid-optimal performance given a sufficient amount of training data and some sufficiently capable and regularized training procedure. This should in theory allow deep learning systems to be able to generalize, since the empirical risk in theory can approximate the true risk arbitrarily well given sufficient training data support.

2.3.2 Understanding Generalisation

As the analysis in 2.2 shows, ERM nonetheless readily fails to generate generalizable predictors with respect to out-of-distribution data). Understanding exactly why this is the case is a subject of ongoing study, and cannot be uniquely attributed to any one property of the machine learning pipeline. In broad strokes, the literature attributes generalization failure to two key properties of modern deep learning pipeline, namely that DNNs readily learn non-robust features, and that DNNs are underspecified by the training data. Naturally, there is some degree of overlap between these two views, and both phenomena induce similar behaviour. In an attempt to systematize the To fully understand the nuances that distinguish the respective arguments, it is useful to first consider the assumptions upon which ERM is based, namely that: TODO: FIX

1. f exists in $\mathcal{H}$
2. R_{emp} is sufficiently sampled
3. $\{x_i, y_i\}$ is an IID sampling of $P(x, y)$. This is the aforementioned iid assumption.
4. $\hat{h}$ is unique in $\mathcal{H}$
5. The optimizer consistently finds $\hat{h}$ (given that it exists and is unique)

As the following sections will show, violations of any one of these assumptions can and typically will result in generalization failure.

2.3.3 Underfitting, Overfitting and Regularization

Violations of assumptions 1 and 2 correspond to well known and fairly well understood forms of generalization failure, namely underfitting and overfitting. One can however argue that these factors can be all but discounted as plausible explanations for the pervasiveness of generalization failure.

Underfitting occurs when the model simply lacks the complexity required to encapsulate the patterns necessary to form generalizable predictions. To give a simple example - consider the problem of trying to fit a linear model to the following dataset: Obviously, no amount of optimization of the parameters in the linear model can ever result in a sufficient description of the underlying data and the function it follows, namely $y = x^2$.

It is commonly accepted that DNNs are not particularly susceptible to underfitting. Modern DNNs, as it turns out, can after all be shown to have infinite effective capacity for practical purposes. It can for instance be shown that even a 2-layer feedforward neural network is capable of fitting noise to random labels with 100% accuracy [44]. Assuming this ability translates to a capacity to generalize (which may not necessarily be the case, as will be discussed in chapter 6), it is fairly reasonable to expect that

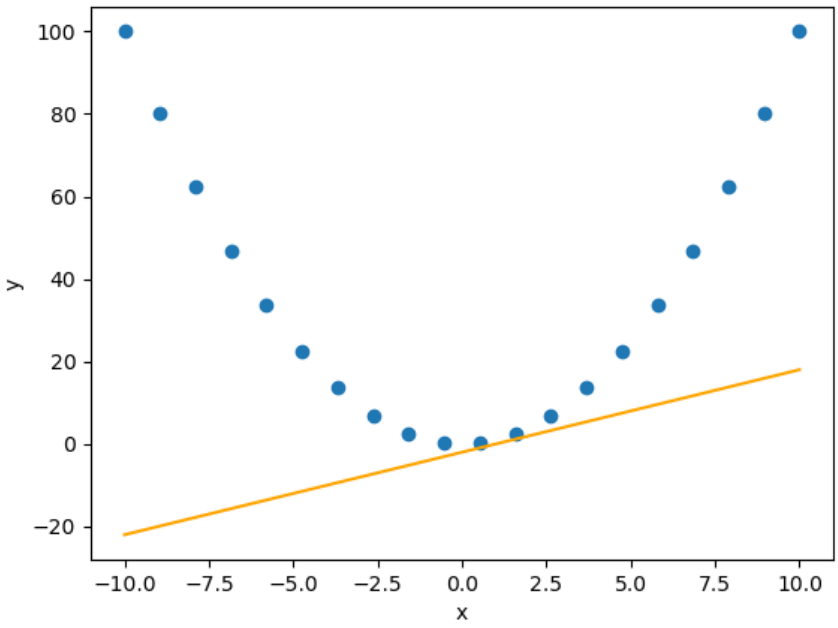

Figure 2.2: A linear model cannot fit polynomial data

the hypothesis space of the highly complex models used today contains a generalizable predictor.

This is also evidenced by the fact that practically all deep learning pipelines take considerable steps towards avoiding the exact opposite of underfitting - overfitting. Overfitting occurs when the model learns patterns that by conventional wisdom are too complex to be likely to actually describe the process(es) that generate them. Once again, to give a somewhat simple example: The underlying function is once again $y = x^2$. The sinusoidal pattern in the interpolated function is obviously erroneous, but this is impossible to determine with complete certainty given only a finite sampling of the underlying function. More generally, every finite sampling of any function can be interpolated as any one of an uncountably infinite set of functions. To prove this, consider the task of finding a function that fits the series $(1, 4, 9)$. Though ones first instinct would be to assume the pattern is described by the function $y = x^2$, and that the next number in the sequence thus is 16, the next number can really be anything at all. Using Newton interpolation, or for that matter any arbitrary interpolation scheme, one can easily find a polynomial that fits the sequence $(1, 4, 9, n, n \in \mathbb{R}$. Since the set of real numbers is uncountably infinite, it follows that the space of functions that fits this new sequence also is uncountably infinite, depending on n. Thus, there is an uncountably infinite number of functions that describe $1, 4, 9$ as well. This of course does not only apply to extrapolating the next element in a sequence, but also any interpolating between consecutive elements. TODO clarify

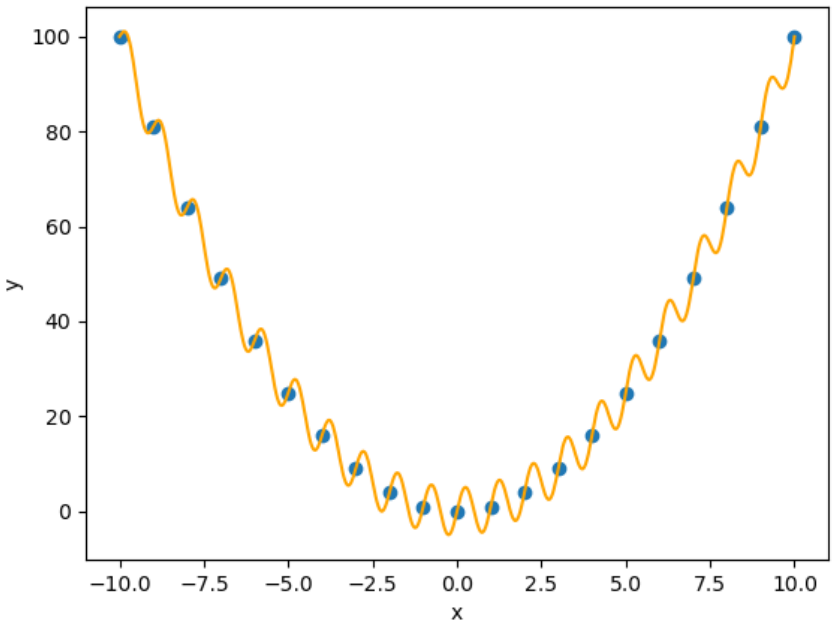

Figure 2.3: A model with excessive capacity interpolates unneccesarily complex patterns

Naturally, this also applies to neural networks. Though instead of discrete sequences, it is probability distributions that are being modelled. Just like how extrapolation and interpolation between points is ill-defined for sequences, it is ill-defined for probability distributions. Consequently, it is necessary to introduce a number of assumptions and constraints to the problem. (CLARIFY)

Obviously, the minimum level of generalization one should achieve is generalization to iid data. To this end, it is necessary to incorporate an iid evaluation method, such as keeping hold-out sets, cross validation, etc. This is of course ubiquitous in modern deep learning. Moreover, it is necessary to make some assumptions regarding the simplicity - or rather, complexity - of the resulting predictor. Certain neurons should for instance not dominate the predictions, the weights should have reasonable values, etc. This is what motivates so-called regularization, whereby different methods - for instance batch normalization, drop-out, L2 loss, weight decay, etc - are used to bias the search towards areas in the loss landscape that are more credible from a perspective of model complexity. This is of course only necessary because assumptions 2 and 4 do not hold. Given a perfect (or at least very well sampled) representation of the risk and consequently the loss-landscape and a perfect optimizer, the model would not overfit in any meaningful way.

Regularization has of course for this reason been extensively studied, and for most purposes the existing regularization techniques suffice just

fine for IID data, and typically yield highly performant predictors so long as it is not subjected to any form of distributional shift. Consequently, Overfitting, though naturally still a factor that needs to be accounted for when designing deep learning pipelines, is not at fault for the generalization failure that is exhibited in modern deep learning systems.

2.3.4 Structural Misalignment and dataset bias

Generalization failure is often attributed to structural misalignment between the predictor as generated by ERM and the causal structure which it ideally should encode [3, 14, 23, 33]. Generally, this misalignment occurs as a result of the predictor learning spurious or otherwise causally unrepresentative features that nonetheless perform well within the training distribution. This if of course made evident as soon as the predictor is exposed to any form of distributional shift, at which point it will (typically) fail to generalize. These distributional shifts can range in magnitude, from changes in imaging modalities, common corruptions such as noise or blurs [17] or spatial transforms [10] to practically imperceptible perturbations, typically exemplified by adversarial attacks [7]. ERM does not and cannot guarantee invariance to distributional shifts, as it assumes that the training data is IID to $R(h)$. This is not, however, necessarily as much of a flaw with ERM inasmuch as it is a flaw in the reasoning behind our expectations.

To illustrate, consider the rather pertinent example of training a model exclusively on either white-light or narrow-band endoscopy. Assume that there are two datasets, each containing samples depicting identical scenes, with the only difference being that dataset A employs white-light endoscopy, whereas dataset B employs narrow-band endoscopy. Ideally, a model trained on either dataset should generate predictors that can generalize to the other, and though one may be optimistic and hope this is the case, this is in no way guaranteed. The causal structure behind the decisions - i.e what exactly makes a polyp a polyp - is never considered at any point in the training process. Instead, the models will simply try to leverage whatever predictive patterns can be found the training data. The model trained on narrow-band images may for instance principally consider the textural characteristic of the polyps, which narrow-band endoscopy enhances. Conversely, the model trained on white-light, lacking access to these textural characteristics, may instead consider more color- or shape-based features. Naturally, if this narrow-band-texture-biased model is deployed in white-light endoscopy, it is not likely to succeed since its principal discriminative features no longer are particularly useful. Similarly, the color-biased model would fail when deployed in narrow-band endoscopy since the colours it once used to distinguish polyps are no longer predictive in narrow-band images.

Though the features each model learns are not particularly representative of the broader context of what makes a polyp a polyp, they make sense when considered from the perspective of either of the two hypothetical modalities. When considering only narrow-band imaging, it makes some sense to heavily weigh the texture of the polyps. When considering only

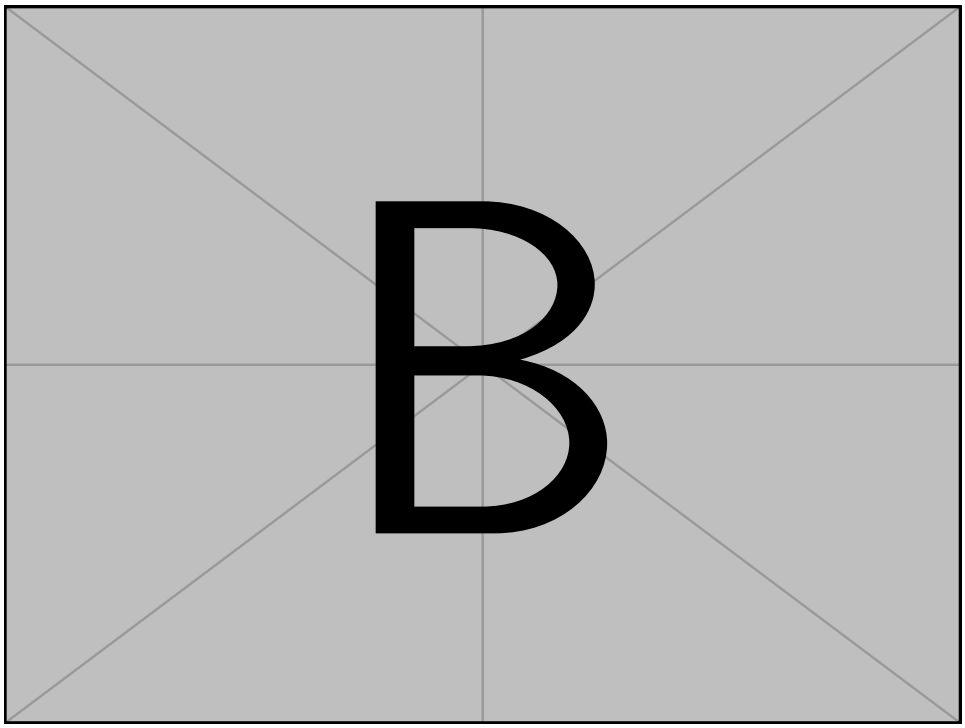

Figure 2.4:

white-light imaging, it makes some sense to heavily weight the shape and colour of the polyps. Though humans are capable of appreciating broader context and subconsciously know that certain features are ancilliary rather than causal (and perhaps more importantly: know the strengths and weaknesses of each modality), DNNs lack the inductive biases needed to take this into account. Once again, DNNs merely leverage the first and best predictive patterns found during the training process, and cannot be expected to optimize for specific invariances on their own, irrespective of how self-evident these invariances may be for humans. This predilection towards dataset-specific features is aptly referred to as dataset bias.

Shortcut Learning

In the aforementioned example, tough the features each predictor learns is not robust to dataset shift, they nevertheless have causal explanations. The causal structure that they correspond to is of course not dataset-agnostic, and as a result flawed in their own way, but the patterns the respective models leverage to interpret the data are not particularly irrational. As it turns out, however, DNNs are unlikely to learn such causally viable features in the first place. In other words, the predictors would not necessarily learn to consider texture in narrow-band images - it could learn any arbitrary pattern so long as it is predictive. Moreover, if such interpretable distributional shifts were the principal cause of generalization failure, generalizability could be induced by explicitly modelling the effects

such shifts induce and taking this into account in the pipeline. In the aforementioned example, one could for instance train some model to map from one lighting environment to the other. Though this would imbue the model with an inherent invariance to the choice of lighting, it is nonetheless not given that the resulting model will be perfectly generalizable.

Consequently, though these detectable forms of distributional shifts also hold some importance when designing generalizable models, a more pervasive and substantially more significant issue is the fact that many of the distributional shifts encountered in clinical settings are not necessarily considered significant or for that matter at all perceptible to a human observer. A human would for instance not be significantly affected by slightly noisy, blurry, rotated, or compressed images, nor would they in all likelihood notice these perturbations. DNNs, on the other hand, have been shown to be highly sensitive to these and several other forms of minor perturbations [17, 18, 22, 35]. Moreover, a human would likely not pick up on the subtle phenotypic cues that may exist in the colon during endoscopy, whereas a DNN may leverage some of these cues to inform their decisions.

It is important to note, however, that despite how these two forms of distributional shift may at surface level appear as completely seperate classes of problems, they can both be traced to the same phenomena - namely that DNNs do not leverage any form of causal logic to inform their decisions and, as mentioned previously, simply exploit any sufficiently predictive pattern they may observe in the data. This phenomenon has been shown to be pervasive across all manner of domains, from natural language processing and computer vision to reinforcement learning and algorithmic decision making. This is referred to as shortcut learning [14] or the Clever Hans effect [24]. Shortcut learning and the brittle features it corresponds to have also been identified as one of the key properties that explains the effectiveness and pervasiveness of adversarial attacks [23]. Naturally, a generalizable predictor should be robust to such minor perturbations, as the model should not in the first place be learning features that get perturbed to any significant degree by adding such high-frequency, low-amplitude noise. Adversarial attacks simply leverage the high degrees of sensitivity inherent to shortcut features, and construct perturbations according the direction in the search space that corresponds to the principal component of this sensitivity [28].

2.3.5 Underspecification

Closely related to shortcut learning is underspecification [8]. A machine learning pipeline can be considered underspecified when it can return any number of risk-equivalent predictors when evaluated on an iid holdout set, dependent only on the random variables used within the training procedure - i.e dropout, seed initialization, and so on. Even with identical hyperparameters, a given training procedure can return any number of predictors each having learned different patterns. One predictor may have learned one shortcut, another may have learned a different shortcut, and one may have fully learned the actual causal relationships it is intended to.

With ERM, and in particular with iid-oriented evaluation procedures, these are all erroneously considered equivalent.

Note that this does not however presuppose anything about the relative occurrence rates of generalizable and non-generalizable predictors. It may not necessarily even be the case that the pipeline can return a generalizable predictor at all. Only that there exists a multitude of risk-equivalent predictors in the search space the optimizer typically explores. Nor does it presuppose anything about the distribution of predictors, only that there is indeed a distribution.

This is evidenced by the fact that generalizability can vary greatly depending on the choice of random seed used during training. In the foundational paper on underspecification in deep learning, for instance, it was highligted that certain classification pipelines can produce predictors that vary in ood accuracy by up to 10% [8]. This is a function of the robustness of the learned features and how likely the pipeline is to return the corresponding predictors.

2.3.6 A probablisitic perspective of generalization

As established, modern deep learning pipelines are not capable of reliably returning generalizable predictors. However, they are not necessarily precluded from it. One can to some extent model this probabilistically by considering the distribution of parameters given the training data, $p(w|\mathcal{D})$. Though it is impossible to know which part of this distribution corresponds to generalizable predictors, it has been shown that marginalizing over this distribution - in other words, bayesian learning - increases generalizability [4, 21, 36, 39].

[39] provides a probabilistic perspective of this phenomenon. They consider generalizability as a two-dimensional quantity, consisting of the support and inductive biases of a model.

2.4 Related work on generalizable deep learning

To summarize, generalization failure occurs due to the weaknesses inherent to ERM. The features that ERM learn to incorporate are often spurious, and the predictors that leverage them are often returned practically arbitrarily from the pipeline. The approaches that have exhibited the highest degrees of success towards increasing generalization as a consequence tend to address these issues in some way or another.

One of the most well-studied approaches to increasing generalizability is the use of data augmentation. Data augmentation is typically implemented in deep learning pipelines in order to prevent overfitting, often in conjunction with other regularization methods. Naturally, overfitting also constitutes generalizability failure in its own right, but augmentation has also been shown to have positive effects for out-of-distribution generalization. It has for instance been shown that carefully designing augmentation procedures increases the generalizability of polyp segmentation models [15]

and prostate segmentation [32]. There has also been a large body of work dedicated to leveraging recent advances in generative models such as generative adversarial networks (GANs) and variational autoencoders (VaEs) to serve as synthetic data augmentation. These types of approaches have also been been shown to increase generalizability in CT segmentation [31] and x-ray based covid detection [27]. Data augmentation facilitates generalization by implicitly biasing the pipeline towards credible inductive biases, as the empirical risk will be best minimized by leveraging features that are conducive to minimizing risk across both synthetic or otherwise augmented data and unaugmented data. Naturally, the choice of augmentations that are used is an inductive bias in and of itself; by employing random rotations, rotational invariance is presupposed. By employing random cropping, image-space object size invariance is presupposed. By employing additive noise, invariance to additive noise is presupposed, and so on. From an ERM perspective, this corresponds to improving the empirical risk estimate; given enough data, all of these invariances may be learned automatically. With a finite and often fairly limited dataset, this may not necessarily be the case simply due to the large number of confounding variables involved.

Another type of approach involves biasing the pipeline towards learning more structured and causally viable latent representations. This is also somewhat well understood when considered through the lens of regularization: drop-out and weight-decay are often employed in order to reduce overfitting under the assumption that a generalizable predictor should not base its decisions on only a few of the available neurons and that seperate neurons should instead encode independent representations of the input. Though there is limited research ont the effects of dropout on ood generalization, constraining the latent representations in DNNs has been shown to be effective method to increase generalizability. For polyp-segmentation it has for instance been shown that adding context-based attention layers to multiple blocks in a given network results in state-of-the-art IoU scores for out-of-distribution evaluation on certain datasets [25]. Other attention-based approaches have also shown promise in this regard [16, 42]. This permits the model to learn and generate attention maps for its latent representations, thus in theory biasing the model towards learning more a more structured interretation of the data.

Multi-task and/or multistage learning has also been leveraged to this end. By jointly optimizing for multiple tasks/subtasks, the model can be biased towards learning features that describe the input data well independent of their performance on any one of the relevant tasks. For poylp-segmentation, for instance, it has been shown that adding image reconstruction as an auxiliary task [37] or decoupling the segmentation task into a coarse segmentation and refinement stage [12] increases generalizability.

More closely supervised methods, wherein certain inductive biases have been explicitly introduced to the pipeline, have also been shown to have some promise. One paper for instance reported an increased robustness to image perturbations after adding a custom filter bank

designed to emulate the primary visual cortex of primates to the front of the CNN [9]. Another reported that models trained on imagenet exhibited significantly higher robustness when explicitly biased towards shape-based features [13].

These approaches all provide workarounds to flaws with ERM, typically to limited practical effect. Consequently, a growing body of work has instead been investigating the idea of foregoing ERM altogether in favor of developing alternative training paradigms. In so-called Invariant Risk Minimization[3], for instance, the model trains to ignore spurious correlations by optimizing for predictors that exhibit stable performance across multiple training environments. A similar approach, namely model-based robust deep learning [30], employs a similar idea in conjunction with distributional modelling. The model is trained such that it is robust to perturbations as generated by a so-called model of natural variation. If this model for instance describes the function mapping one training environment to another, this will then optimize for predictors that are invariant to the distributional shift this function corresponds to.

Finally, Bayesian learning has also been shown to improve generalization. In particular, ensemble-based networks - which mathematically can be considered an approximation of Bayesian marginalization [38, 39] - have demonstrated high degrees of generalizability for polyp-segmentation [21, 36]. Since deep learning pipelines are typically underspecified, one can consider an ensemble of predictors to be a sampling of the distribution over parameters given training data. These predictors are of course unlikely to have learned identical representations, and consequently whatever spurious correlations inferred by one predictor may be countered by the features employed in another. Consequently, features that are stable across predictors and consequently more likely to generalize are weighed to a greater extent, in turn mitigating the effects of underspecification. Structural misalignment may nevertheless impact ensembles, and as such the effective returns are limited.

Chapter 3

Methods and Implementation

Summarizing the key points made in chapter 2, generalization is in large part dependent on the causal robustness of the features that the model learns. The problem of achieving generalization, then, boils down to imposing constraints to the deep learning pipeline such that the model ignores non-causal patterns.

Directly discriminating between causal and non-causal patterns is, however, somewhat intractable. For one, the patterns that neural networks learn are often difficult to identify, and even more difficult to understand semantically. Though the field of explainable AI has made significant progress in this regard, there simply is not a way to determine the causality of whatever correlation DNNs infer. Moreover, establishing this causality necessitates a complete understanding of the problem the model is trying to reason about in the first place. If this was at all possible, one might as well use the knowledge required to do so to design a classifier using conventional image-analysis.

Though establishing what is causal is difficult, establishing what *isn't* causal is not all that complicated. To give a concrete example, consider once again the problem of classifying images of cows and camels. Associating the cow class with grass and the camel class with sand is obviously non-causal, since this pattern would not hold if the model for instance is asked to detect cows on Mars or camels on the Moon. To mitigate this, one may simply collect data of cows and camels in differing backgrounds, but such careful curation of datasets is not typically feasible. A better approach is to instead simply augment the data. In particular, one can generate multiple instances of the same sample but with varied backgrounds. Consistency across this augmented set can then be rewarded in order to bias the model towards learning background-invariant features.

This, of course, applies to more than just modifying backgrounds: the more of these non-causal changes to the input data- from this point on referred to as perturbations - are accounted for and modelled, the more spurious correlations are excluded from the search, and the more likely the model is to learn the patterns that are actually causal. After all, a pattern can for all intents and purposes be considered causal when it holds when subjected to all forms of perturbations.

Thus, though rewarding causal behaviour is intractable, punishing non-causal behaviour is not. All that is required to do so is to be able to apply perturbations that highlight the non-causal reasoning the model employs, quantify the model's sensitivity to these perturbations, then minimize this quantity through optimization. The resulting model will then have learned invariance to whatever causally irrelevant information that the perturbation defines. This property of being invariant to perturbations will be referred to as the consistency of the model.

Thus, consistency is in effect a surrogate for generalizability; if a model is consistent, it is invariant to non-causal patterns, and if it is invariant to non-causal patterns, it necessarily employs causal patterns. Optimizing for consistency can as a result mitigate both shortcut-learning and underspecification, subject only to the span of the space of perturbations and how well inconsistent behaviour can be quantified. For instance, if the perturbations affect the image such that certain shortcuts are broken, these shortcuts are less likely to be learned. A similar argument can be made for underspecification: if multiple predictors are risk equivalent but nevertheless encode conflicting inductive biases, probing the inductive biases learned by the respective predictors through various perturbations can reveal which are generalizable and which are not.

Naturally, this all presupposes that there is some model that can output all possible perturbations one might desire the model to be invariant to. This is of course not the case. As highlighted by the pervasiveness of adversarial attacks and the relative ineffectiveness of adversarial defences, the perturbations that break DNNs are not necesssarily intuitive and are difficult to analyze in a manner that is conducive to the task of engineering invariance thereto.

Nevertheless, much stands to be gained if the model learns to be invariant even to a fairly limited space of perturbations. Though generalizability is by no means guaranteed in this case, the odds of learning generalizable features are improved simply because the perturbations limits the types of patterns that a given model can learn. If for instance a white-light endoscopic image is perturbed such that it mimics a narrow-band image, and the model learns to be invariant to this perturbation, predictors that leverage lighting-dependent features will no longer be returned from ERM.

This approach, then, requires two components: a perturbation model, and some form of training procedure that imposes invariance to the transformations that the perturbation model employs - or in other words, learns to be consistent. To this end, a numeric representation of consistency in the context of segmentation needs to be established.

3.1 Segmentation Inconsistency Coefficient

In the context of segmentation, consistency is the ability of the model to output a reasonable segmentation when the input data is subjected to some perturbation. Inconsistency, by the same token, is the extent to which

the output changes when subjected to a perturbation beytond what is reasonable. One simple approach to express this numerically would simply be to count the number of pixels that change given some perturbation, and normalize this with respect to the total number of pixels, or in other words calculating the intersection over union, also known as the Jaccard Index, across the perturbed and unperturbed segmentations. However, the ground truth may of course change as a result of the perturbation - if the image is rotated, for example, the segmentation mask should be rotated accordingly. If an image is globally distorted in some way, the segmentation should exhibit the corresponding distortion. If an image is exposed to low-amplitude additive noise, the segmentation should not really be affected at all, and so on. This, of course, all needs to be taken into account. This can be acheived by discounting the proportion of pixels that is expected to change from the overall metric. Formally, this metric, from this point referred to as the segmentation Inconsistency coefficient (SIC) can be expressed as follows:

Let $Y := \{y, \hat{y} := f(x)\}$ be the set consisting of the segmentation labels (masks) and predictions for the unperturbed samples, where $f(\cdot)$ denotes the model. Let $\epsilon(\cdot)$ be some perturbation function. Then, let $A := \{a := \epsilon(y), \hat{a} := f(\epsilon(x))\}$ be the set consisting of segmentation predictions and masks when the input is subjected to a perturbation. Inconsistency can then be defined by:

$$L_c = \frac{1}{\sum\{y \cup a \cup \hat{y} \cup \hat{a}\}} \sum\{y \ominus \hat{y} \ominus a \ominus \hat{a}\} \tag{3.1}$$

Where $\ominus$ denotes the symmetric difference, or in terms of boolean algebra, the XOR operator. This, as mentioned, corresponds to counting the number of pixels that change after the input is subjected to a perturbation - $\hat{a} \ominus \hat{y}$, but discounting those we expect to change, $a \ominus y$. It should be noted that this metric is minimised not only if the predictions are both correct and consistent with one another, but also if the predictions are both incorrect, so long as whatever change that occurs is consistent with the expected change. This is illustrated in Figure 3.1

Note, however, that this metric does not presuppose what transformation has occurred. In Figure 3.1, for instance, the change induced by the perturbation may correspond to simply moving the polyp in the image (and replacing the empty space with a believable background), or it may correspond to a rotation by 90 degrees. How this should be analysed with respect to consistency is up to interpretation - one can argue that a rotation should rotate the incorrect predictions as well, or one can argue that it should only rotate the correct component of the prediction. For simplicity, SIC is based on the latter interpretation.

3.2 Consistency Loss

Similarly to how one can optimize for the Jaccard Index through the Jaccard Loss, so too can one optimize for consistency by using SIC as a component

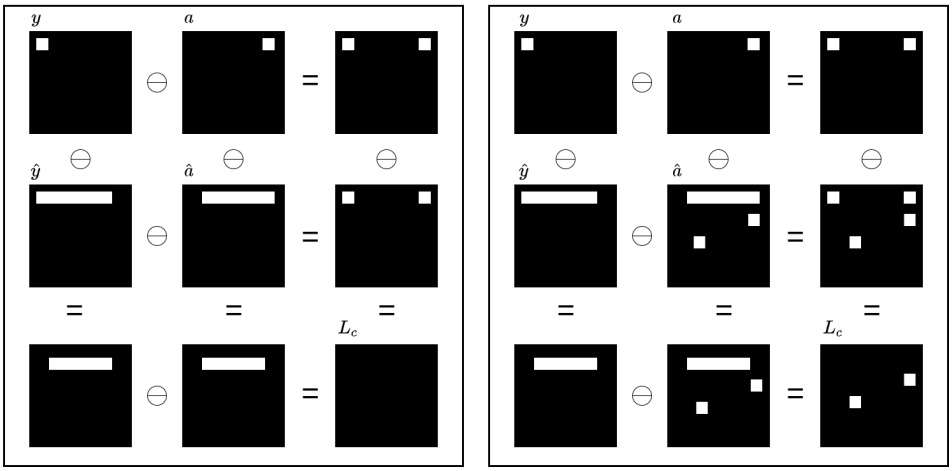

Figure 3.1: Visualisation of SIC sets, where white is a positive prediction. Note that loss is zero regardless of prediction correctness so long as it changes in the expected manner. Note also that the symmetric difference operators are associative. Left shows an instance of consistent but partially incorrect predictions, and right shows an instance of inconsistent but partially correct predictions

of a loss function. Naturally, using SIC on its own is not really useful since it only expresses consistency, and is wholly agnostic to whatever object it is trying to segment. Consequently, it has to be combined with a some conventional segmentation loss. A simple way to do this would be to simply add them together and normalize, i.e:

$$L(Y, A) = L_{seg}(Y) + L_c(Y, A)$$

Preliminary experiments showed that this, however, exhibited some degree of instability. The model would readily get stuck in local minima where its predictions were indeed consistent, but also consistently predicting artifacts. Examples of this can be found in the appendix. To mitigate this, a better loss function involves adaptively weighing the respective components according to the segmentation performance. This way, the model will learn generally correct interpretations early in the training, then start weighing consistency more and more as the model sees improvements to its segmentation performance:

$$L = (1 - IoU) \times L_{seg} + IoU \times L_c \tag{3.2}$$

If the Jaccard Index (IoU) is used, this is also equivalent to:

$$L = {L_{jac}}^2 + (1 - L_{jac}) \times L_c \tag{3.3}$$

Using this formulation, the model will principally be focusing on learning consistent inductive biases at the end of the training procedure, and correct itself if this starts to bias the search away from sufficiently performant regions.

3.3 Perturbation Models

So far, it has been assumed that a perturbation model has been given beforehand. This is of course not the case, and naturally any such model needs to be designed with respect to the domain in question. Rotational invariance makes sense for endoscopic images, for instance, but not for classification of hand-written numbers. Thus, in order to engineer such a model, it is first necessary to establish what invariances are desired for the given task. In the case of polyp-segmentation, it is clear that it is necessary to account for variability in for instance lighting, image-resolution, polyp-size, polyp-shape, polyp-location, camera-quality, color-shifts, blurs, optical distortions, and affine transformations. Thus, a model is required that can (more or less) parametrize this variability. Broadly speaking, these transformations can be categorized as follows:

- Pixel-wise variability, which affect only the image, i.e color-shifts, brightness shifts, contrast-shifts, lighting, blurs etc
- Geometric variability, which affect both the image and the segmentation mask by some parametrizable quantity, i.e affine transforms and distortions
- Manifold variablity, which affects both the image and the segmentation mask depending on a learned model of the distribution, i.e the size, shape and location of polyps

Pixel-wise variability and geometric variability can be modelled fairly trivially through the use of the same transformations typically used in conventional data-augmentation. Manifold-variability, however, is somewhat more difficult, and requires a functional representation of the distribution. [30] and [31] acheive this through cross-dataset style-transfer, but this of course necessitates multiple datasets. Given only one dataset, a different method must be used. For a classification task, this could for instance be DeepAugment (cite) or a similar technique, but since the perturbations need to be fairly well defined and easily explainable, it is necessary to be able to model how such augmentations affect the ground truth segmentation mask. To this end, a GAN-inpainter can be used. The inpainter can be trained to generated a polyp given a region mask, which can then be overlaid with the original image.

Gan-based polyp inpainting

As mentioned in Chapter 2, the use of GANs and other distributional modelling in the context of generalization is typically restricted to image-to-image translation, and typically involve transforming an image drawn from one distribution such that it is iid with a second distribution. This, though interesting and no doubt useful assuming such datasets are readily available, has limited practical use. It is not necessarily always the case that there exists multiple datasets depicting identical problems, and merely translating between modalities does not as mentioned earlier in the

thesis ensure generalizability. This thesis instead aims to investigate how generalizable a predictor can be made given only a single dataset.

3.3.1 Geometric and pixel-wise transformations

3.4 PLACEHOLDER ALGORITHM NAME

Consistency Training

Adversarial Consistency Training

3.5 Baselines and Generalizability Metrics

3.5.1 Baseline Models

3.5.2 Performance Metrics

3.5.3 Datasets

3.6 Implementation details

3.7 Experiments

3.7.1 MNV-testing

3.7.2 Training methods

Chapter 4

Results

Chapter 5

Analysis

5.1 Augmentation Robustness and Consistency Loss

As the results show, the performance of the pipeline that merely used augmentations is more or less equivalent to the performance exhibited by the modified pipeline. There is a very good reason for this: Consistency loss is mathematically equivalent to data augmentation, up to the choice of hyperparemeters - i.e augmentation probability, learning rates, etc. This section presents a proof of this fact, along with a theoretical analysis of how data augmentation affects the pipeline.

5.1.1 Data augmentation

Let $Y := \{y, \hat{y} := f(x)\}$ be the set consisting of the segmentation predictions and masks for the unaugmented samples, and $A := \{a := MNV(y), \hat{a} := f(MNV(x))\}$ be the set consisting of segmentation predictions and masks for the augmented samples. Finally, let $Z := \{z, \hat{z}\} \in_R \{Y, A\}$. The loss function subject to data augmentation can then be expressed as $L(Z \in_R Y, A)$, where L is any loss function. For the sake of simplicity in remaining calculations, this will be treated as the Jaccard loss, i.e $L(y, \hat{y}) := 1 - \sum y \cap \hat{y} / \sum y \cup \hat{y}$

$$L(Z \in_R Y, A)$$

5.1.2 Consistency loss

$$L_s = \frac{1}{\sum \hat{y} \cup \hat{a}} \sum \hat{y} \ominus y \tag{5.1}$$

$$L_c = \frac{1}{\sum \hat{y} \cup \hat{a}} \sum [\hat{y} \ominus y \ominus \hat{a} \ominus a] \tag{5.2}$$

$$L_{c+s} = L_c(Y, A) + L_s(Y) \tag{5.3}$$

$$= \frac{1}{\sum \hat{y} \cup \hat{a}} \left[\sum \{\hat{y} \ominus y \ominus \hat{a} \ominus a\} + \sum \{\hat{y} \ominus y\} \right] \tag{5.4}$$

$$\begin{aligned} &= \frac{1}{\sum \hat{y} \cup \hat{a}} \Bigg[\sum \{\hat{y} \ominus y\} + \sum \{\hat{a} \ominus a\} \\ &- \sum \{\backslash y \cap \hat{y} \cap a \cap \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap a \cap \hat{a}\} \cup \\ &\{y \cap \hat{y} \cap \backslash a \cap \hat{a}\} \cup \{y \cap \hat{y} \cap a \cap \backslash \hat{a}\} \\ &- \sum \{\backslash y \cap \hat{y} \cap \backslash a \cap \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap \backslash a \cap \hat{a}\} - \\ &\cup \{\backslash y \cap \hat{y} \cap a \cap \backslash \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap a \cap \backslash \hat{a}\} \\ &+ \sum \{\hat{y} \ominus y\} \Bigg] \end{aligned} \tag{5.5}$$

$$\begin{aligned} &= 2L_s(y, \hat{y}) + L_s(a, \hat{a}) + \frac{1}{\sum \hat{y} \cup \hat{a}} \Bigg[\\ &- \sum \{\backslash y \cap \hat{y} \cap a \cap \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap a \cap \hat{a}\} \cup \\ &\{y \cap \hat{y} \cap \backslash a \cap \hat{a}\} \cup \{y \cap \hat{y} \cap a \cap \backslash \hat{a}\} \\ &- \sum \{\backslash y \cap \hat{y} \cap \backslash a \cap \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap \backslash a \cap \hat{a}\} - \\ &\cup \{\backslash y \cap \hat{y} \cap a \cap \backslash \hat{a}\} \cup \{y \cap \backslash \hat{y} \cap a \cap \backslash \hat{a}\} \Bigg] \end{aligned} \tag{5.6}$$

The non-loss terms in equation 5.6 are proper subsets of the symmetric difference of the mask and segmentation across either dataset. The component of the loss that corresponds to these terms consequently grows in proportion to both $L_s(y, \hat{y})$ and $L_s(a, \hat{a})$. L_{c+s} and L_{sy+sa} are therefore monotonically correlated - i.e, when one grows, the other grows with it, and when one falls, the other one falls with it.

5.1.3 Adversarial Dice

$$L = \frac{1}{2} L(a, \hat{a}) + \frac{1}{2} L(y, \hat{y})$$

This should be asymptotically equivalent to data augmentation wit p=0.5

Chapter 6

Discussion

6.1

6.2 Auxilliary findings

asdasdf

Bibliography

[1] Sharib Ali et al. 'EndoCV 2021 3rd International Workshop and Challenge on Computer Vision in En-doscopy'. In: ().

[2] Sharib Ali et al. 'Preface to: EndoCV2020Computer Vision in Endoscopy'. In: *CEUR Workshop Proceedings*. Vol. 2595. CEUR Workshop Proceedings. 2020.

[3] Martin Arjovsky et al. *Invariant Risk Minimization*. 2020. arXiv: 1907.02893 [stat.ML].

[4] Razieh Baradaran and Hossein Amirkhani. 'Ensemble learning-based approach for improving generalization capability of machine reading comprehension systems'. In: *Neurocomputing* 466 (2021), pp. 229–242. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2021.08.095. URL: https://www.sciencedirect.com/science/article/pii/S0925231221012923.

[5] Ishita Barua et al. 'Artificial intelligence for polyp detection during colonoscopy: a systematic review and meta-analysis'. en. In: *Endoscopy* 53.3 (Mar. 2021), pp. 277–284.

[6] Emma Beede et al. 'A Human-Centered Evaluation of a Deep Learning System Deployed in Clinics for the Detection of Diabetic Retinopathy'. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. Honolulu, HI, USA: Association for Computing Machinery, 2020, 1–12. ISBN: 9781450367080. DOI: 10.1145/3313831.3376718. URL: https://doi.org/10.1145/3313831.3376718.

[7] Battista Biggio et al. 'Evasion Attacks against Machine Learning at Test Time'. In: *Lecture Notes in Computer Science* (2013), 387–402. ISSN: 1611-3349. DOI: 10.1007/978-3-642-40994-3_25. URL: http://dx.doi.org/10.1007/978-3-642-40994-3_25.

[8] Alexander D'Amour et al. *Underspecification Presents Challenges for Credibility in Modern Machine Learning*. 2020. arXiv: 2011.03395 [cs.LG].

[9] Joel Dapello et al. 'Simulating a Primary Visual Cortex at the Front of CNNs Improves Robustness to Image Perturbations'. In: *bioRxiv* (2020). DOI: 10.1101/2020.06.16.154542. eprint: https://www.biorxiv.org/content/early/2020/06/17/2020.06.16.154542.full.pdf. URL: https://www.biorxiv.org/content/early/2020/06/17/2020.06.16.154542.

[10] Logan Engstrom et al. *Exploring the Landscape of Spatial Robustness*. 2019. arXiv: 1712.02779 [cs.LG].

[11] Ivan Evtimov et al. 'Robust Physical-World Attacks on Machine Learning Models'. In: *CoRR* abs/1707.08945 (2017). arXiv: 1707.08945. URL: http://arxiv.org/abs/1707.08945.

[12] Adrian Galdran, Gustavo Carneiro and Miguel A. González Ballester. 'Double Encoder-Decoder Networks for Gastrointestinal Polyp Segmentation'. In: *Lecture Notes in Computer Science* (2021), 293–307. ISSN: 1611-3349. DOI: 10.1007/978-3-030-68763-2_22. URL: http://dx.doi.org/10.1007/978-3-030-68763-2_22.

[13] Robert Geirhos et al. *ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness*. 2019. arXiv: 1811.12231 [cs.CV].

[14] Robert Geirhos et al. 'Shortcut learning in deep neural networks'. In: *Nature Machine Intelligence* 2.11 (2020), 665–673. ISSN: 2522-5839. DOI: 10.1038/s42256-020-00257-z. URL: http://dx.doi.org/10.1038/s42256-020-00257-z.

[15] Raman Ghimirea, Sahadev Poudelb and Sang-Woong Leec. 'An Augmentation Strategy with Lightweight Network for Polyp Segmentation'. In: (2021).

[16] Ran Gu et al. 'Domain Composition and Attention for Unseen-Domain Generalizable Medical Image Segmentation'. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*. Ed. by Marleen de Bruijne et al. Cham: Springer International Publishing, 2021, pp. 241–250. ISBN: 978-3-030-87199-4.

[17] Dan Hendrycks and Thomas Dietterich. *Benchmarking Neural Network Robustness to Common Corruptions and Perturbations*. 2019. arXiv: 1903.12261 [cs.LG].

[18] Dan Hendrycks and Thomas Dietterich. *Benchmarking Neural Network Robustness to Common Corruptions and Perturbations*. 2019. arXiv: 1903.12261 [cs.LG].

[19] D Heresbach et al. 'Miss rate for colorectal neoplastic polyps: a prospective multicenter study of back-to-back video colonoscopies'. en. In: *Endoscopy* 40.4 (Apr. 2008), pp. 284–290.

[20] Steven Hicks et al. 'The EndoTect 2020 Challenge: Evaluation and Comparison of Classification, Segmentation and Inference Time for Endoscopy'. In: Feb. 2021, pp. 263–274. ISBN: 978-3-030-68792-2. DOI: 10.1007/978-3-030-68793-9_18.

[21] Ayoung Honga et al. 'Deep Learning Model Generalization with Ensemble in Endoscopic Images'. In: (2021).

[22] Hossein Hosseini, Baicen Xiao and Radha Poovendran. *Google's Cloud Vision API Is Not Robust To Noise*. 2017. arXiv: 1704.05051 [cs.CV].

[23] Andrew Ilyas et al. *Adversarial Examples Are Not Bugs, They Are Features*. 2019. arXiv: 1905.02175 [stat.ML].

[24] Jacob Kauffmann et al. *The Clever Hans Effect in Anomaly Detection*. 2020. arXiv: 2006.10609 [cs.LG].

[25] Taehun Kim, Hyemin Lee and Daijin Kim. 'UACANet: Uncertainty Augmented Context Attention for Polyp Segmentation'. In: *Proceedings of the 29th ACM International Conference on Multimedia*. 2021, pp. 2167–2175.

[26] A Leslie et al. 'The colorectal adenoma-carcinoma sequence'. en. In: *Br. J. Surg.* 89.7 (July 2002), pp. 845–860.

[27] Mohammad Momeny et al. 'Learning-to-augment strategy using noisy and denoised data: Improving generalizability of deep CNN for the detection of COVID-19 in X-ray images'. In: *Computers in Biology and Medicine* 136 (2021), p. 104704. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2021.104704. URL: https://www.sciencedirect.com/science/article/pii/S0010482521004984.

[28] Roman Novak et al. *Sensitivity and Generalization in Neural Networks: an Empirical Study*. 2018. arXiv: 1802.08760 [stat.ML].

[29] D K Rex et al. 'Colonoscopic miss rates of adenomas determined by back-to-back colonoscopies'. en. In: *Gastroenterology* 112.1 (Jan. 1997), pp. 24–28.

[30] Alexander Robey, Hamed Hassani and George J. Pappas. *Model-Based Robust Deep Learning: Generalizing to Natural, Out-of-Distribution Data*. 2020. arXiv: 2005.10247 [cs.LG].

[31] Veit Sandfort et al. 'Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks'. In: *Scientific Reports* 9 (Nov. 2019). DOI: 10.1038/s41598-019-52737-x.

[32] Thomas H. Sanford et al. 'Data Augmentation and Transfer Learning to Improve Generalizability of an Automated Prostate Segmentation Model'. In: *American Journal of Roentgenology* 215.6 (2020). PMID: 33052737, pp. 1403–1410. DOI: 10.2214/AJR.19.22347. eprint: https://doi.org/10.2214/AJR.19.22347. URL: https://doi.org/10.2214/AJR.19.22347.

[33] Bernhard Schölkopf. *Causality for Machine Learning*. 2019. arXiv: 1911.10500 [cs.LG].

[34] Dinggang Shen, Guorong Wu and Heung-Il Suk. 'Deep Learning in Medical Image Analysis'. In: *Annual Review of Biomedical Engineering* 19.1 (2017). PMID: 28301734, pp. 221–248. DOI: 10.1146/annurev-bioeng-071516-044442. eprint: https://doi.org/10.1146/annurev-bioeng-071516-044442. URL: https://doi.org/10.1146/annurev-bioeng-071516-044442.

[35] Ashish Shrivastava et al. 'Learning From Simulated and Unsupervised Images Through Adversarial Training'. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[36] Vajira Thambawita et al. *DivergentNets: Medical Image Segmentation by Network Ensemble*. 2021. arXiv: 2107.00283 [eess.IV].

[37] Nikhil Kumar Tomar et al. *DDANet: Dual Decoder Attention Network for Automatic Polyp Segmentation*. 2020. arXiv: 2012.15245 [eess.IV].

[38] Andrew Gordon Wilson. *The Case for Bayesian Deep Learning*. 2020. arXiv: 2001.10995 [cs.LG].

[39] Andrew Gordon Wilson and Pavel Izmailov. *Bayesian Deep Learning and a Probabilistic Perspective of Generalization*. 2020. arXiv: 2002.08791 [cs.LG].

[40] Sidney J. Winawer et al. 'Prevention of Colorectal Cancer by Colonoscopic Polypectomy'. In: *New England Journal of Medicine* 329.27 (1993). PMID: 8247072, pp. 1977–1981. DOI: 10.1056/NEJM199312303292701. eprint: https://doi.org/10.1056/NEJM199312303292701. URL: https://doi.org/10.1056/NEJM199312303292701.

[41] Julia Winkler et al. 'Association Between Surgical Skin Markings in Dermoscopic Images and Diagnostic Performance of a Deep Learning Convolutional Neural Network for Melanoma Recognition'. In: *JAMA Dermatology* 155 (Aug. 2019). DOI: 10.1001/jamadermatol.2019.1735.

[42] ChengHui Yua, JiangPeng Yana and Xiu Lia. 'Parallel Res2Net-based Network with Reverse Attention for Polyp Segmentation'. In: (2021).

[43] John R. Zech et al. 'Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study'. In: *PLOS Medicine* 15.11 (Nov. 2018), pp. 1–17. DOI: 10.1371/journal.pmed.1002683. URL: https://doi.org/10.1371/journal.pmed.1002683.

[44] Chiyuan Zhang et al. *Understanding deep learning requires rethinking generalization*. 2017. arXiv: 1611.03530 [cs.LG].