Universidade Estadual de Campinas
Instituto de Computação

Levy Gurgel Chaves

Evaluating Self-Supervised Pre-Training in Out-of-Distribution and Low-Data Scenarios

Avaliação de Modelos Auto-Supervisionados para Cenários Fora da Distribuição e com Poucos Dados

CAMPINAS
2022

Levy Gurgel Chaves

Evaluating Self-Supervised Pre-Training in Out-of-Distribution and Low-Data Scenarios

Avaliação de Modelos Auto-Supervisionados para Cenários Fora da Distribuição e com Poucos Dados

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Dissertation presented to the Institute of Computing of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Computer Science.

Supervisor/Orientadora: Profa. Dra. Sandra Eliza Fontes de Avila
Co-supervisor/Coorientador: Prof. Dr. Eduardo Alves do Valle Junior

Este exemplar corresponde à versão final da Dissertação defendida por Levy Gurgel Chaves e orientada pela Profa. Dra. Sandra Eliza Fontes de Avila.

CAMPINAS
2022

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

C398e

Chaves, Levy Gurgel, 1997-
Evaluating self-supervised pre-training in out-of-distribution and low-data scenarios / Levy Gurgel Chaves. – Campinas, SP : [s.n.], 2022.

Orientador: Sandra Eliza Fontes de Avila.
Coorientador: Eduardo Alves do Valle Junior.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Aprendizado profundo. 2. Aprendizado de máquina. 3. Aprendizado auto supervisionado (Aprendizado do computador). 4. Melanoma. 5. Visão por computador. I. Avila, Sandra Eliza Fontes de, 1982-. II. Valle Junior, Eduardo Alves do. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Avaliação de modelos auto-supervisionados para cenários fora da distribuição e com poucos dados
Palavras-chave em inglês:
Deep learning
Machine learning
Self-supervised learning (Machine learning)
Melanoma
Computer vision
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Sandra Eliza Fontes de Avila [Orientador]
Hélio Pedrini
Cristina Nader Vasconcelos
Data de defesa: 17-03-2022
Programa de Pós-Graduação: Ciência da Computação

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0002-7431-2440
- Currículo Lattes do autor: http://lattes.cnpq.br/1403509513761670

Universidade Estadual de Campinas
Instituto de Computação

Levy Gurgel Chaves

Evaluating Self-Supervised Pre-Training in Out-of-Distribution and Low-Data Scenarios

Avaliação de Modelos Auto-Supervisionados para Cenários Fora da Distribuição e com Poucos Dados

Banca Examinadora:

- Profa. Dra. Sandra Eliza Fontes de Avila
  Universidade Estadual de Campinas
- Profa. Dra. Cristina Nader Vasconcelos
  Google Brain
- Prof. Dr. Hélio Pedrini
  Universidade Estadual de Campinas

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Campinas, 17 de março de 2022

Acknowledgments

I thank UNICAMP for the unique opportunities and memories of the last years. In special, I am very thankful to the Artificial Intelligence Lab. (recod.ai), our lab at the Institute of Computing (IC/UNICAMP).

I gratefully thank QuintoAndar for the Master's scholarship. This study was also financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Finally, I thank Microsoft, NVIDIA, and grant 2017/16246-0, São Paulo Research Foundation (FAPESP) for the all infrastructures investments, which enabled this research.

Resumo

O aprendizado auto-supervisionado estreitou relações entre o aprendizado supervisionado e o não supervisionado. O primeiro leva a modelos mais precisos, mas requer amostras anotadas por humanos, enquanto o segundo explora amostras não anotadas, mas muitas vezes leva a precisões decepcionantes. Usando anotações sintetizadas nas chamadas tarefas pretexto, a auto-supervisão possibilita pré-treinar modelos em abundantes pseudo-rótulos antes de ajustá-los para a tarefa alvo. Este trabalho avalia e compara cinco pré-treinamentos auto-supervisionados em relação ao tradicional método de referência totalmente supervisionado para tarefas de classificação de imagens médicas e naturais. Consideramos casos desafiadores quando há apenas poucos dados — 1% e 10% do conjunto de treinamento original — disponíveis e quando os conjuntos de teste contiverem mudanças desconhecidas na distribuição dos dados (fora da distribuição). Cobrimos uma gama de quatro tarefas de classificação médica: câncer de pele, câncer de mama, tumor cerebral e amostras histopatológicas, e duas tarefas de propósito geral: classificação de animais e veículos. Nossos resultados sugerem que o pré-treinamento supervisionado na ImageNet é preferível em cenários de poucos dados e fora da distribuição quando as classes da tarefa alvo são conhecidas no momento do pré-treinamento, ou seja, quando as classes alvo são um subconjunto do subconjunto das classes original de pré-treinamento. Em aplicações médicas, os desempenhos de modelos auto-supervisionados variaram muito. Mesmo com cada modelo em seu ápice em performance, nenhum dos cinco métodos auto-supervisionados investigados se mostrou consistentemente melhor do que o modelo supervisionado de referência em todas as aplicações alvo, mas o melhor variou de acordo com a tarefa alvo. Entretanto, se há pouca ou nenhuma diferença entre métodos supervisionados e auto-supervisionados considerando o desempenho, então o treinamento auto-supervisionado pode ser preferível pois elimina a necessidade de dados rotulados na etapa de pré-treinamento. Além disso, o aprendizado auto-supervisionado o possibilita que o pré-treinamento original seja continuado usando exemplos rotulados e não rotulados da tarefa alvo antes da etapa de ajuste fino. Presumimos que tal comportamento ocorra devido ao pré-treinamento auto-supervisionado original que acrescenta dificuldade em capturar detalhes de baixa variação interclasse e intraclasse em aplicações médicas.

Abstract

Self-supervised learning bridges the gap between supervised and unsupervised learning. The former leads to the most accurate models but requires human-annotated samples, while the latter exploits non-annotated samples but often leads to disappointing accuracies. By using synthesized annotations on so-called pretext tasks, self-supervision can pre-train models on abundant pseudo-labels before tuning them for the downstream (target) task. This work assesses five self-supervision schemes against a supervised baseline for medical and natural image classification tasks. We consider challenging cases when low-data samples — 1% and 10% of the original training set — are available and test sets contain unknown distribution shifts (out-of-distribution) at training time. We cover a range of four medical classification tasks: skin cancer, breast cancer, brain tumor, and histopathology samples, and two general-purpose tasks: animal and vehicle classification. Our results suggest that supervised pre-training on ImageNet is preferable in low-data and out-distribution scenarios when the classes of the target task are known at pre-training time, i.e., when the target classes are a subset of the original one. In medical applications, self-supervised pre-training performances varied a lot. Even with each model at its peak, none of the five investigated self-supervised methods have proven consistently better than the supervised baseline in all target applications, but the best one varied depending on the target task. However, if there is no or little difference between supervised and self-supervised methods in performance, then self-supervised training may be preferable because it eliminates the need for labeled data in the pre-training step. Moreover, self-supervised learning makes it possible to continue the original pre-training using labeled and unlabeled data from the target task before the fine-tuning step. We hypothesize that such behavior occurs due to the original self-supervised pre-training scheme adding difficulty in capturing the low inter-class and intra-class variation details in medical applications.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Learning predictive, high-level, and discriminative visual representations without labels is one of the desired goals of Machine Learning. By relying only on unlabeled data, such a requirement allows accessible data widely available on the internet, such as images, texts, and audios, to train machine learning models. Supervised methods usually lead to better results in standard computer vision tasks due to the explicit human-annotated data expressing the relationship between samples of the same class. However, providing label annotations is time-consuming and costly. On the other hand, unsupervised methods learn by extracting visual correspondences without labels, causing a drop in performance compared to supervised methods. The model needs to explicitly extract semantic correlations from the data without supervision about the underlying classes. It would be valuable to explore other types of supervised signals rather than human-provided labels to train those models to learn high-level visual features since human labels are hard to obtain for all computer vision tasks. Here is where **self-supervised learning** takes its place.

"*Basically, it's the idea of learning to represent the world before learning a task. This is what babies and animals do. We run about the world, we learn how it works before we learn any task. Once we have good representations of the world, learning a task requires few trials and few samples.*"[1] — Yann LeCun, the director of Facebook AI and one of the most influential scientists in the Artificial Intelligence field. Humans require little to no supervision to learn to interact with their surroundings. The condensed representations they learn can generalize to several tasks, and they can easily assemble observations from past experiences and a wide variety of senses, such as sound, touch, and vision. A desirable property is to build models that learn meaningful visual representations only from observations. This gap has prompted efforts to bring similar behavior into Machine Learning models. A desirable property is that intelligent models have autonomy and do not rely only on explicit labels to learn new visual concepts, and the acquired knowledge is helpful for various objectives and tasks.

The core behind self-supervised learning is to take the voluminous of available unlabeled data and use it to understand the world by itself. Self-supervision allows machines to comprehend the part of a data sample and find out which part is missing or, by contrast, similar versions of the same image by framing a supervised approach using only unlabeled data. This allows models to learn autonomously, enabling them to learn general concepts, such as shapes, colors,

[1] https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence

object dynamics, and high-level concepts with high transferability capabilities.

Deep Neural Networks (DNNs) proved to excel in most standard computer vision tasks [74]. Deeper DNN models typically lead to better performance, but at the cost of massive computational resources needed to train state-of-the-art methods and datasets with millions of labeled examples are a vital part of their tremendous success. However, collecting and annotating large-scale datasets for many different tasks is infeasible. The underlying network's representations might be biased to a specific geometric position of the objects or task, depending on how they were collected [109]. Researchers have recently started investigating alternative ways to train neural networks without explicitly labeled data to diminish the burden of depending only on labeled data. These methods, commonly referred to *self-supervised learning* [67], have become a powerful tool for large-scale machine vision. Currently, self-supervised learning and *unsupervised learning* terms are being used interchangeably in the literature.

Also, self-supervised learning appears as a promising learning paradigm to improve the generalization capabilities of machine learning models [23]. The cornerstone of many learning algorithms is to assume the data distribution between training and test are independent and identically distributed (iid) [54], and they would suffer when this condition is unsatisfied. However, it is an unrealistic constraint to guarantee real-world applications: cases that violate the data distribution assumption are pretty standard. For example, different institutions usually acquire medical images in clinical practice with distinct imaging protocols, ranging from diverse patient populations to scanner vendors. Such protocols add difficulty for machine learning models by introducing distribution shifts commonly absent in training data. A model learned in one data distribution (training set) could be applied in many other distributions (several test sets). This case draws attention to machine learning models' performances in environments with unknown distribution shifts.

The performance in many environments under several distribution shifts raises concerns about the generalization capacity of trained machine learning models. The usual method of measuring the model's generalization is evaluating a single test set drawn from the same distribution as the training set. However, this protocol provides only a naive in-distribution performance guarantee: a small test error indicates a similar performance on new samples drawn from the same distribution as the training set. It is unfeasible to train a model on the exactly known data distribution in training in many applications. A model will face out-of-distribution data on which its performance will change a lot compared to in-distribution performance.

The core objective of this Master Dissertation is to evaluate self-supervised pre-trained models' performance in medical and general-purpose applications under distinct scenarios that machine learning practitioners would probably encounter in real-world applications. We cover challenging scenarios where the test sets range from in- and out-of-distribution samples and low-data regimes[2]. We cover a range of four medical classification tasks: skin lesion screening, breast cancer, brain tumor, and histopathology samples; and two general-purpose tasks: animal and vehicle classification. We performed two evaluation protocols for each scenario based on the low-data regime and out-of-distribution performance. The former aims to evaluate the performance when only a few percentages (1% and 10%) of the original training set are available.

[2]We use the term low-data instead of few-shot [120] evaluation because the latter requires that each class has the same number of samples. We do not follow such restrictions in our protocol. We still face a scenario where only a few data samples are available, so we use the low-data nomenclature.

We designed a scarce data evaluation, and it is particularly desirable in medical applications since only a few data sampled are labeled if compared to standard computer vision datasets. The latter focuses on assessing how the trained supervised and self-supervised models perform on out-of-distribution test datasets. We highlight the importance of such protocol to assess generalization capabilities being crucial to avoid spurious correlations on both medical [9, 12] and general-purpose applications [42] at inference time.

We started our investigation engaging in skin cancer classification due to the extensive experience our research group at the forefront of top-tier research in skin lesion analysis [8–13, 40, 41, 83, 94, 95, 97, 111]. According to the rapid development of the self-supervised learning field in 2020, we were curious to explore how these new models would perform in skin lesion analysis and if they are competitive with their supervised counterparts. Our two batches of experiments show it is advantageous to employ self-supervised learning, especially in low-data regimens. Low-data experiments intend to understand how the model's performance is affected when only a small subset (1% and 10%) of the original training is available. In general trends, we also observed self-supervised learning is more robust to out-of-distribution scenarios — we test the trained model on several distribution-shifted test sets keeping the same task (number of classes) as in training.

A reasonable and natural way to extend our previous experiments is to verify if the same behavior occurs in other medical applications and general-purpose classification tasks. Our results suggest that self-supervised pre-training methods are slightly superior in performance with low-data and out-of-distribution samples. Interestingly, we found it advantageous to use supervised pre-training when the fine-tuned classes in the target task are a subset of those in pre-training. No single self-supervised pre-training scheme dominates in all tasks, implying that a universal pre-training scheme remains a mystery. Although self-supervised gains are small, we look at it positively: self-supervised allows us to perform an in-domain unsupervised pre-training, and it can serve as an alternative to supervised pre-training for transfer learning tasks.

1.1 Research Questions

The key research questions that permeate this Master Dissertation are:

Q1. Is there any benefit in using self-supervised models instead of supervised models as a starting point for fine-tuning?

Q2. How do self-supervised models pre-trained on ImageNet perform in medical imaging compared to supervised pre-trained ImageNet models?

Q3. How do self-supervised models perform when only a few samples are available for training and when out-of-distribution test datasets for medical and general-purpose applications?

1.2 Contributions

The main contributions of this work are:

C1. We assessed the influence of five self-supervised pre-training schemes (BYOL [48], InfoMin [107], MoCo [52], SimCLR [23], and SwAV [17]) versus fully supervised approaches in several medical and general-purpose classification tasks;

C2. We designed challenging in- and out-of-distribution testing scenarios to assess the model's generalization capabilities for medical and general-purpose image classification tasks;

C3. We surveyed and organized several databases freely available in the medical literature to set up a challenging scenario to assess the out-of-distribution performance of the trained models. We covered several different medical imagining modalities with at least one extra dataset to measure the robustness of out-distribution performance;

C4. We assessed the performance of both self-supervised and supervised models in a low-data training scenario — when there is only 1% and 10% of the original training set available;

C5. Part of this Master Dissertation is available in the ArXiv platform with the name "An Evaluation of Self-Supervised Pre-Training for Skin-Lesion Analysis" [19]. The code for reproduce part of this work is available at `https://github.com/VirtualSpaceman/ssl-skin-lesions`

1.3 Outline

We organized the remainder of this work as follows. In Chapter 2, we review the fundamental concepts of self-supervised learning and contrastive learning — an idea that boosted the self-supervised field. In Chapter 3, we review the literature on self-supervised methods for general-purpose and medical applications, emphasizing the recent generative and predictive approaches, along with their promising ideas that made the computer vision community draw attention to self-supervised learning. In Chapter 4, we describe the pipeline, the methodology, and the experimental design to perform all the evaluations for medical and general-purpose applications. In Chapter 5, we detail the datasets, metrics, and results covering all investigated applications, highlighting the performance in the full- and low-data scenario. Also, we test the trained model's generalization capabilities at specially crafted out-distribution sets, which aim to mimic real distribution shifts in real-world applications. Finally, in Chapter 6, we summarize and analyze our findings and address the limitations of this work. We also indicate future directions for the approached problems.

Chapter 2

Related Concepts

This chapter introduces fundamental related concepts to understand this Master Dissertation. First, we explain self-supervised learning, covering how it works, how to evaluate it, and the current challenges. Next, we present the idea of contrastive learning, the critical ingredient that empowers self-supervision. Finally, we detail all evaluated self-supervised methods we used in the experiments.

2.1 Self-Supervised Learning

One way to characterize *self-supervised learning* is by comparing it to unsupervised learning. They both aims to learn data representation without any annotation, but the former aims to remove the time-consuming human annotation by creatively exploring some properties in the data to set up a supervision task. The motivation is relatively straightforward. Creating a dataset with clean labels is expensive, but the internet continually generates unlabeled data. One way to leverage that amount of unlabeled data is to set the learning objectives properly to frame a supervision problem from the data itself.

The core idea lies in designing an auxiliary *pretext task* (or self-supervised task) that provides supervision to pre-train neural networks. The pretext task is designed to answer the following question: "How can we design such a task to explore some property inherent from our data to learn robust feature representations with high transferability capabilities to several tasks?". Developing such tasks is the heart of the field, and there is no recipe.

A straightforward example of a pretext task for images is to predict image rotations. The task involves randomly rotating images in the dataset by 0, 90, 180, or 270, and the target objective is to predict the rotation label [44]. To perform well in this task, the model should learn a latent feature representation that can efficiently discriminate all orientations of different objects. The rotation prediction task is made-up, so the actual accuracy is unimportant [74], like how we treat auxiliary tasks. However, at the end of the pre-training task, we expect the latent feature representations contain enough semantic information to boost transfer performance for other tasks.

Figure 2.1 shows the general pipeline for self-supervised learning. The pipeline resembles the idea of transfer learning, in which we commonly take advantage of pre-trained models trained in a supervised way to better performance at target tasks. The performance on the pre-

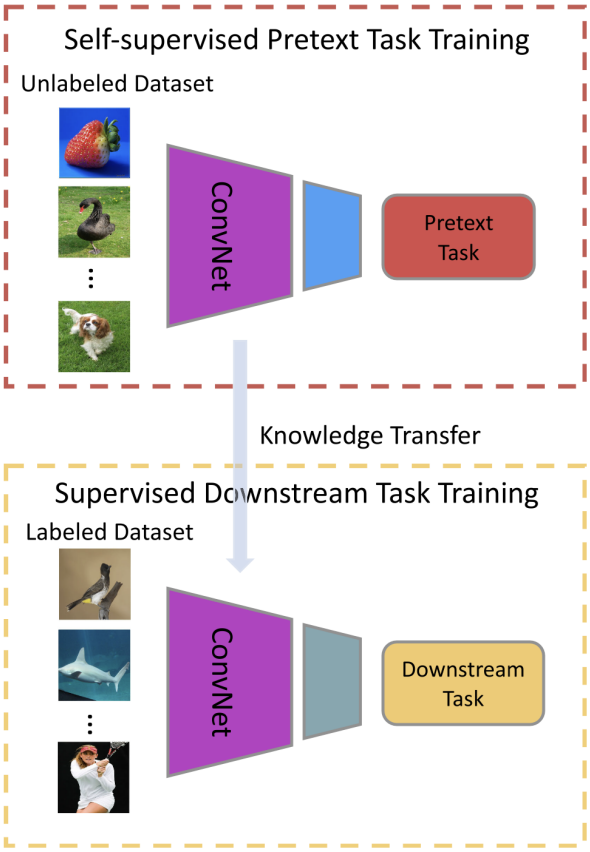

Figure 2.1: The general pipeline of self-supervised learning. The visual features are learned by solving a designed pretext task. The learned representations can be further transferred to downstream (target) classes when the self-supervised pre-training is complete. Figure reproduced from Jing et al. [67].

text task is usually not so important [74] and the learned representations are evaluated based on their performance in other tasks known as downstream tasks. For downstream tasks, we obtain the visual feature representations for new images from the intermediate feature representations of the pre-trained model. In other words, the model acts as an image encoder that transforms high-dimensional pixel data into high-level semantic feature vectors. The mainstream protocol in classification tasks is to assess the resulting feature representation by training a linear classifier on top of a frozen encoder (neural network). Linear classifiers are preferred because they only rely on the representations' quality and discriminative power. Other approaches involve using the pre-trained encoder as initialization for other tasks such as object detection, semantic segmentation, and segmentation.

Self-supervised learning methods have integrated both generative and context-based approaches. Generative methods involve generating or synthesizing new data. Numerous works focus either on the idea of autoencoders [115] or adversarial networks [45]. Unlike generative methods, the context-based leverages some context properties available in the data, e.g., the spatial structure in images or word order in text data. Their adoption rapidly drew the attention of the machine learning community, especially when *contrastive learning* scheme appeared as a pre-training alternative.

2.2 Contrastive Learning

Contrastive learning empowered self-supervised learning by introducing an effective manner to pre-train models with unsupervised data. The concept is quite old (first appeared in 2005 [30]), and only in 2020 did it regain steam, with minor modifications compared to its original formulation. The core of contrastive learning is to generate lower-dimensional image representations such that similar instances are close to each other and far from dissimilar ones (see Figure 2.2).

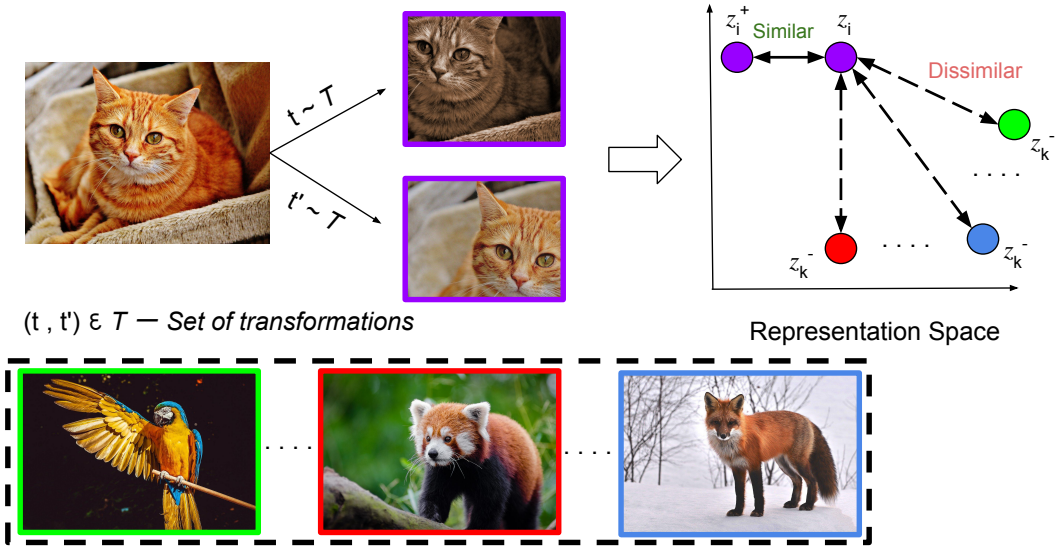

Figure 2.2: Visual representation of the contrastive learning idea. Given an anchor image (cat), we sample a positive pair based on data augmentation and construct the negative pairs based on random samples in the dataset. Essentially, the loss encourages representations of positive pairs (in purple) to be close to each other in representation space while making representations of negative pairs dissimilar apart (in green, red, and blue). Best viewed in color.

To explicitly encourage the model to contrast correctly, the contrastive setting uses the idea of generating image pairs based on three key concepts: anchor, positive, and negative(s) representations. The *anchor* is a reference sample, the *positive* one belongs to the same class or shares some semantic as the anchor (purple boxes in Figure 2.2), and the *negative* one (green, red, and blue boxes in Figure 2.2) is a sample that does not belong to the same category as the anchor. If the label information is available, we could make positive image pairs by simply exploring the label information and then organizing the representation space to cluster samples that share the same label information. Many approaches might be taken to generate positive and negative samples, but, as far as we do not have any class label information, the core idea is to assume that each data sample belongs to its class [35, 122]. The most common way is: given an anchor image x_i, the positive sample (or positive view) x_i^+ is obtained by applying stochastic augmentations in x_i. The negative sample (or negative view) x_i^- is sampled via the same process as positives but augmenting a randomly chosen sample from the dataset. Selecting good positive and negative pairs is still an open question regarding contrastive methods [107].

The self-supervised contrastive tasks involve model loss computation in the latent space by contrasting latent representations of positive and negative samples given a specific context. Learning an encoder function that maps an input image into a representation without explicit

supervision is challenging. Based on the InfoNCE loss [30] (NCE stands for Noise-Contrastive Estimation), Oord et al. [91] managed to adapt it to best fulfill the self-supervised learning necessity to avoid label information. The adapted version is a softmax-based loss as follows:

$$\mathcal{L} = \frac{-1}{2N} \sum_{i=1}^{2N} \log \frac{\exp(sim(z_i, z_i^+))/\tau}{\sum_{k \neq i}^{2N} \exp(sim(z_i, z_k))/\tau}, \tag{2.1}$$

where $z_j = f_\theta(x_j)$ is a normalized anchor latent vector representation of input image x_j parametrized by neural network f_θ with parameters θ, z_j^+ and z_j^- are, respectively, positive and negative samples, $sim(\cdot, \cdot)$ is any function that computes the similarity between two vectors, $\tau > 0$ is a scalar temperature hyperparameter. Originally the dot product was taken as main distance function. Note that for each anchor j exists one positive pair and $2N - 2$ negative samples.

2.3 Evaluated Methods

We describe five self-supervised methods (SimCLR [54], MoCo [26], BYOL [48], InfoMin [107], and SwAV [17]) we evaluate in this Master Dissertation.

SimCLR (Simple framework for Contrastive Learning of visual Representations) proposes an end-to-end learning [54] (see Figure 2.3). A random set of data augmentations (flipping, color distortion, Gaussian noise) are applied to the input image, resulting in a pair of correlated views (positive pair). Both views are passed to a shared encoder to obtain their latent representation. The representations are given to a projection head to project them into a low-level latent dimension that is much smaller than the original space. These projections are used to calculate the InfoNCE [91] loss and maximize the mutual information between positive pairs, and diminish the mutual information among negative pairs. Their main contribution was to introduce heavy data augmentations and large batch sizes, especially because they rely on in-batch negative sampling. Their work further reduces the gap between self-supervised and supervised learning. For transfer learning over 12 general-purpose image datasets, SimCLR outperforms supervised networks on five datasets.

Two designs in SimCLR are the key to high transfer performance [23]: MLP projection head and heavy data augmentation. MoCo-V2 [26] (Momentum Contrast) combined these two designs, achieving even better transfer performance with no dependency on a considerable batch size (see Figure 2.4).

Grill et al. [48] made a more radical step by removing the need for negative pairs. Historically, negative pairs are crucial in a contrastive learning setting to avoid model collapse and trivial representations. In BYOL [48] (Bootstrapping Your Own Latents), one slow network creates targets for a fast network. The parameters of the fast network are learned by backpropagation, and the parameters of the slow network are the exponential moving average of the parameters of the fast network. In that manner, BYOL bootstraps its own target representations. BYOL still matches data-augmented views between positive pairs as a pretext but without resorting to negative pairs. Instead, it feeds one view to the fast and the other to the slow network and uses the cosine distance between the two outputs as a loss. Figure 2.5 depicts the whole scheme. BYOL uses two neural networks: the target and an online network. The online network

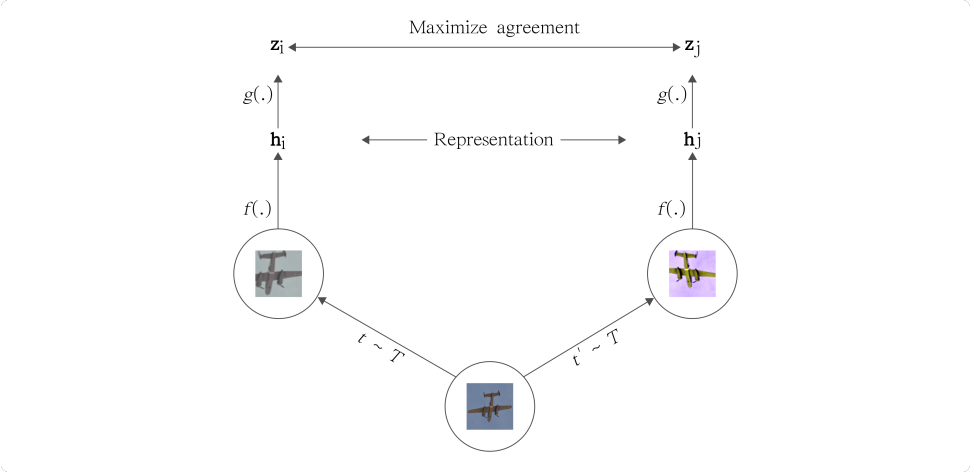

Figure 2.3: SimCLR pre-training scheme. First, they apply a set of heavy data augmentation on the input image to generate the views and the positive pair. Each view is given to a shared encoder $f(.)$ to generate the latent representation h_j. A projection network $g(.)$ is included to project the representation to a low-level embedding. Finally, they maximize the agreement between the positive pairs and encourage the representation of negative pairs to be far from each other. Figure reproduced from Chen et al. [23].

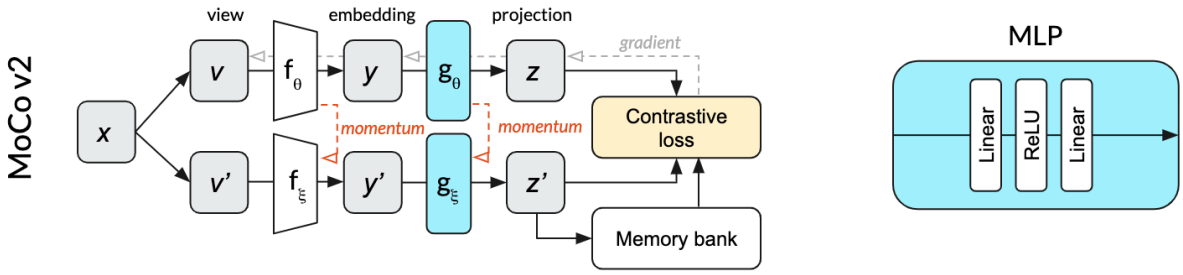

Figure 2.4: MoCo-V2 pre-training scheme. MoCo-V2 splits the single shared network into two sub-networks: online (top row) and momentum (bottom row). The online network is updated by SGD, while the momentum network is updated according to an exponential moving average of the online network weights. MoCo-V2 uses a memory bank of past projections as negative examples for contrastive learning to mitigate sampling in-batch negatives. Figure reproduced from https://generallyintelligent.ai/blog/2020-08-24-understanding-self-supervised-contrastive-learning.

is defined by a set of weights θ, and comprises a set of three stages: an encoder f_θ, a projection network g_θ, and a prediction network q_θ. The target network contains only the first two stages of the online network (encoder and projection). They both have the same architecture but differ in weights.

InfoMin [107] investigated alternative manners to create positive and negative views for contrastive learning. The authors claim that the best views have less mutual information for augmentation-based views in contrastive learning (see Figure 2.6). The views should only share the primary content information (label) in the optimal condition. They first propose an unsupervised method to minimize mutual information between views to produce optimal views. However, this may result in a loss of information for predicting labels (such as a pure blank

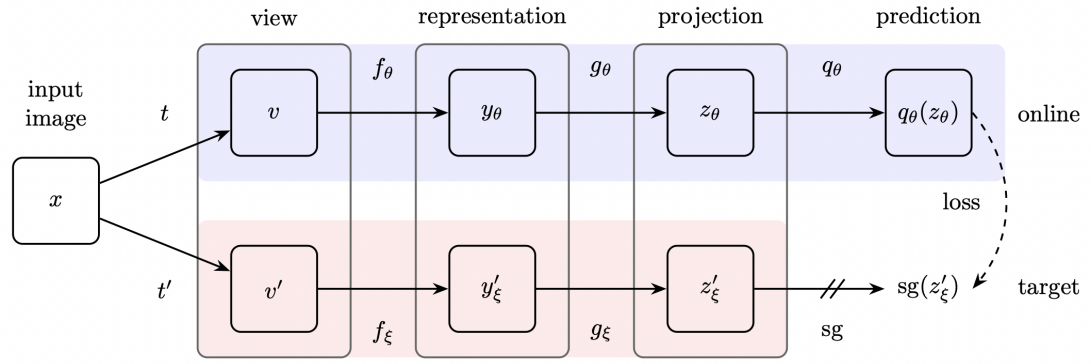

Figure 2.5: BYOL uses the online-target network as in MoCo but adds an extra MLP to the online network. Also, BYOL uses the ℓ_2 error between the target network's normalized prediction and projected representation. Such learning dynamics remove the need for negative samples. Figure reproduced from Grill et al. [48].

view). Consequently, they propose a semi-supervised method to find views sharing only label information.

SwAV [17] (Swapping Assignments Between Views) is a clustering self-supervised pretraining based on clustering (Section 3.1.2 , which introduced an online cluster assignment approach based on learnable prototypes. Cluster assignment is achieved by assigning the latent representations to a set of learned prototype vectors and passing it through the Sinkhorn-Knopp algorithm. They propose a swapped prediction problem where the code (generated by the prototype vectors) of the view of an image is predicted from the representation of another view of the same image (Figure 2.7). The rationale here is that if two views are semantically similar, their codes should also be similar. They also proposed a data augmentation policy referred to as multi-crop. This policy indicates that the same image is randomly cropped to get a pair of high resolution as a global view image and cropped to get additional views of low-resolution images like a local view.

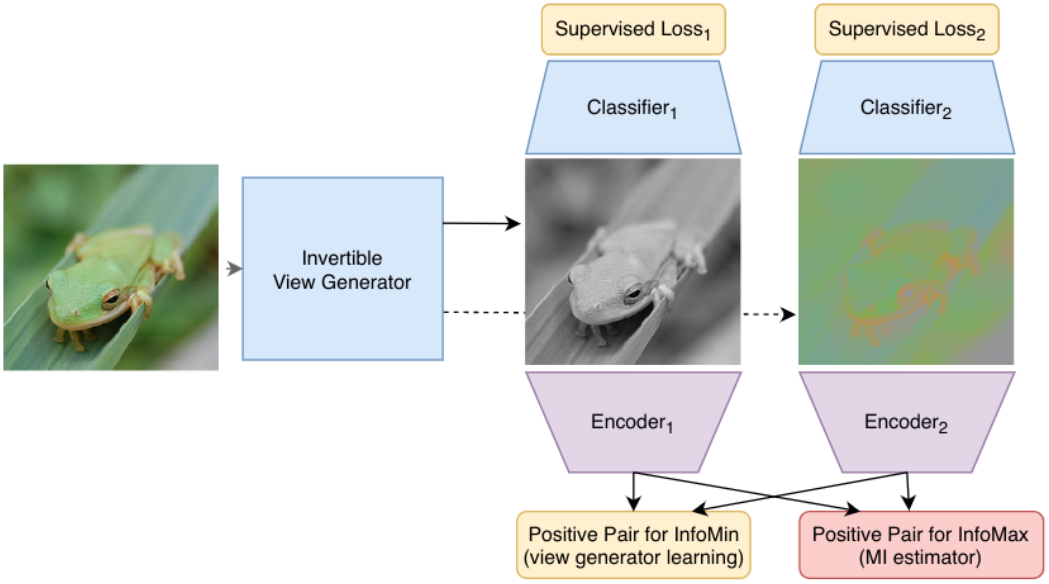

Figure 2.6: InfoMin [107] view generator. The input image is split into two different images (views) using an invertible view generator. To learn the view generator, they minimize the information between views (yellow box) while classifying the object from each view. They train both encoders to maximize the InfoNCE lower bound (red box). They train a neural network with the fixed view generator without the additional supervised classification losses when the view generator finishes training. Figure reproduced from the InfoMin paper [107].

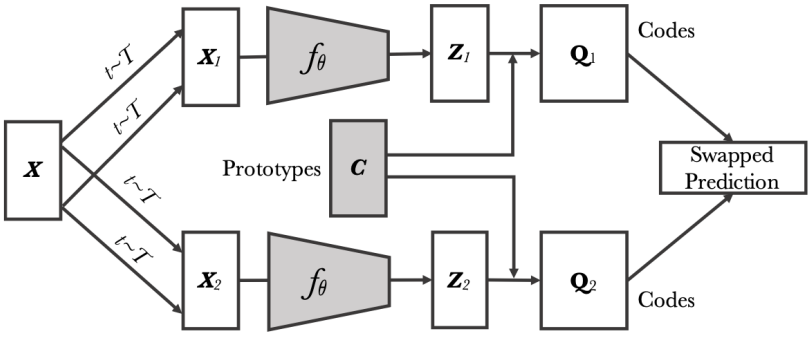

Figure 2.7: SwAV pre-training scheme. It assigns a code vector (prototypes) for each input image based on image feature representation. They train a model to solve a swapped prediction problem, where they predict the "code" vector from one data augmented view from another view. The prototypes are learned jointly with the encoder parameters. Figure reproduced from Caron et al. [17].

Chapter 3

Related Work

In this chapter, we review the literature on self-supervised learning. Here, we only consider self-supervision applied to images and classification tasks. Still, the following ideas are unrestricted to images, and some of them can be highly adaptable for other contexts, such as texts [80] and audio [102]. We conduct our study by first presenting the self-supervised learning methods for visual and medical tasks. In the following sections, we overview how self-supervised learning is categorized. It is composed of two main groups: generation-based (Section 3.1.1) and context-based (Section 3.1.2), in which we provide a comprehensive notion of how the scenario evolved through the years. Such categorization might change as new methods appear.

3.1 Self-Supervised Learning for Visual Tasks

The self-supervised methods can be categorized regarding the data attributes used to design the pretext task. We follow the categorization of Jing et al. [67] with minor modifications due to new updates in the literature. They categorize the pretext tasks in two different branches: generative-based and context-based. Here, we review prior work in each category and how the field evolved over the years. Our criterion of inclusion-exclusion was image-based papers focusing on learning representations in a self-supervised way followed by an evaluation of the learned features on ImageNet [34] or CIFAR [76] datasets with image classification as the downstream task.

3.1.1 Generative-Based Self-Supervision

Generative-based self-supervised methods aim to learn image representations by generating or synthesizing images. The core idea is to teach neural networks to encode all the information about the image on its intermediate representation to perform the generative task.

Suppose a generative model is good at generating a high-quality observation of an image unseen at training time. In that case, this is evidence that it has learned representations that capture the data's spatial structure, such as object locations and object semantics.

Autoencoders [57], mainly used for dimensionality reduction, is the pioneer in the generative-based methods. It imposes a bottleneck in the network forcing a compressed representation of the original input into a low-dimensional vector. Then, the compressed vector goes through an uncompressing stage (decoder network) to reconstruct the original input.

Generally, generative-based methods follow a similar idea but with different pipelines to learn visual features. Pathak et al. [93] introduced the idea of context autoencoder in which the model is trained to predict a corrupted part of the image based on its surroundings (Figure 3.1). The input consists of an RGB image after a binary mask is applied. The boolean value 1 indicates to keep the pixel value of all channels as it is, otherwise turning their value to 0. Thus, the objective is to reconstruct the missing part by minimizing the reconstruction (ℓ_2 norm) and adversarial losses [45] to improve visually appealing outputs.

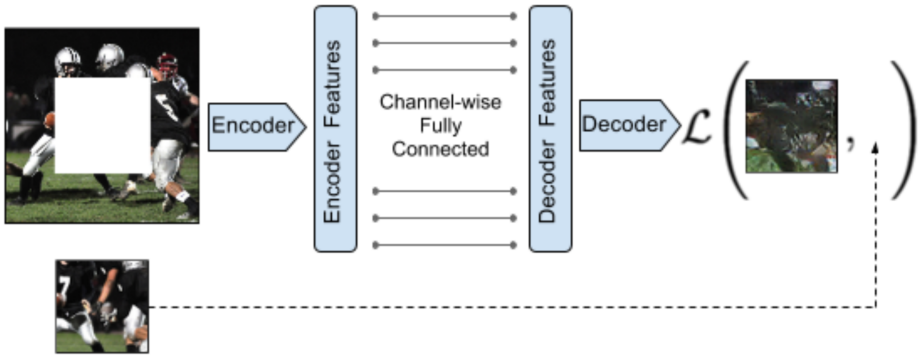

Figure 3.1: Illustration of the context encoders. The input image is masked out (white box at the center) and used as input for a neural network. The pretext task is to reconstruct the missing part given the surrounding of the masked part (context). Figure reproduced from Pathak et al. [93].

In many cases, the semantics of a scene or textures from several objects give cues about the world. For example, the sky is typically blue, and trees are green. Those priors do not work for every object, e.g., horses can be black or brown, but possibly not purple or green. To be aware of such a range of colors, the observer must recognize which object is being observed.

Based on such a concept, Zhang et al. [124] used as a pretext task the idea of hallucinating a plausible colorization for grayscale images to possibly fool the human observer. They trained a model to hallucinate colors given a grayscale image as input. The model receives the lightness channel L from CIE L*ab* colorspace and outputs the corresponding a and b channels. Instead of finding the a and b values for every pixel using a straightforward approach, the objective is to predict a color based on a predefined set of colors. The authors introduced the color rebalancing term in the final loss function to produce more vibrant colors according to their rarity. Also exploring color as pretext tasks, Zhang et al. [125] proposed to predict one subset of data channels from another. They used two disjoint sub-networks for each channel subset to predict the missing channel subset. We refer to Larson et al. [77] for a closer look into colorization as a pretext task as a vehicle for representation learning.

Still, using the dual-network architecture, Jenni et al. [65] trained a neural network to spot synthetic artifacts. First, they trained an autoencoder to reproduce the input image, then the input image is projected in a latent representation, and some features are masked, corrupting its embedded information. A repair neural network is introduced for inputting features through each decoder layer to help the decoder decompress the damaged representation. Therefore, the model is trained jointly to distinguish between fake or real samples, and to output the mask

applied to the latent representation, reminiscent of the concept of Generative Adversarial Networks (GANs) [45].

Using the concept of GANs, Chen et al. [24] introduced the task of predicting rotation angle into the discriminator. Jointly with the standard adversarial loss (predict fake *versus* real), the additional task tries to predict how many degrees the original input was rotated among the following angles {0, 90, 180, 270}. They train the discriminator to detect image rotation angles based only on the real data, preventing the generator from generating images that easily predict the rotation. Other approaches explore adding context-based self-supervision tasks (Section 3.1.2) as additional level of difficulty into discriminator, such as feature matching between latent representations [118], feature exchanging [62] and augmentation prediction [20].

Chen et al. [22] pre-trained generative models by designing a pretext task relying on the original pixel values of the input image. Inspired by the Transformer language models [113], the authors resized raw images to a low resolution and reshaped them into text-like sequences of pixels. They leveraged two pre-training objectives to achieve pixel prediction: autoregressive and masked prediction. The former consists of a sequence of P pixels, and the objective is to predict the same P pixels but shifted by 1 pixel. In the latter approach, a random percentage of the pixels is masked (dropped), and the model needs to output which pixels were dropped.

3.1.2 Context-Based Self-Supervision

Context-based methods perform the learning process by solving pretext tasks designed based on attributes of the context images. Neural networks must understand the content of the entire image and produce a plausible latent representation to encode such rich information. To accomplish this task, the neural network needs to learn and leverage spatial context information such as the relative positions of different parts of an object, the shape of the objects, or the semantics in similar images.

There are three ways of learning from context-based methods as a supervised signal: predictive, contrastive learning, and clustering. In *predictive tasks*, we usually train a neural network to predict a pseudo label of the data based on clustering [16, 17, 112] or explore spatial information [35, 44]. In *contrastive learning*, the task relies on representation learning and tries to directly minimize a distance function between samples from the same group and maximize the feature distance from samples from other groups. *Clustering*-based methods intend to cluster the data, in which similar samples within the same cluster share some semantic information, and the representations are minimized in representation space.

Predictive Learning

Images contain rich spatial context information. Such spatial property is leveraged to design several pretext tasks for self-supervised learning. For example, pretend an auxiliary task of rearranging image patches. We take two random patches of an image and change their position with each other. To solve the task, the first step is to identify which patches were changed and then put each patch into its original arrangement — a high-level understanding of the spatial context information such as the shape of the objects and the relative positions of different parts of an object. Usually, the task involves predicting some “fake” pretext label.

The idea was first introduced in Dosovitskiy et al. [36], where they applied multiple random transformations to an image and used all modifications from the same image to create a surrogate class. They derived a class label (surrogate class) according to the image index in the dataset. For example, if the dataset is composed of $5,000$ images, then will exist $5,000$ classes. Then, they trained a neural network in a supervised manner to classify each data-augmented sample in one of the surrogate classes.

By exploring spatial image context, Doersch et al. [35] formulated the pretext task of predicting the relative position between two image patches (see Figure 3.2). First, a reference patch of the input image is selected at random (a blue patch in Figure 3.2). Considering that the reference patch is placed in the middle of a 3×3 grid, a second patch is sampled from its eight neighboring locations around it. To avoid easily solving the task, noise is added by exploring texture continuity and local level patterns, such as adding gaps and small jitter to neighbor patches. Finally, the model is trained to predict which one of eight neighboring locations selected the second patch.

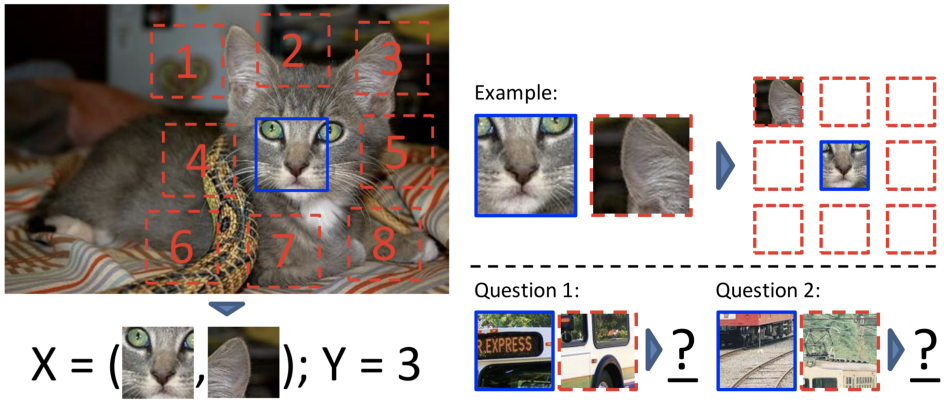

Figure 3.2: Relative position pretext task. Two random patches are extracted and the model is trained to predict the relative position of the second patch taking the first as reference (bounded in blue). Figure reproduced from Doersch et al. [35].

Several approaches leverage the spatial cues for self-supervised representation learning [71, 90, 121]. One typical work is proposed by Noorozi and Favaro [89]. They designed a pretext task to solve Jigsaw Puzzles, mainly based on image tiles (Figure 3.3). Nine disjoint patches are sampled from the input image, similar to relative position prediction. Next, they shuffle as patches and give them as input for a neural network. However, for nine patches, there are a high number of permutations (9! = 362, 880). To alleviate the large solution space, the authors limited the number of permutations into a predefined set based on hamming distance. A natural way to extend this puzzle is by adding extra levels of difficulty, such as damaged patches [71], noisy patches [90], and patch grouping [121]. The main principle of designing puzzle tasks is finding a reasonable task that is neither too difficult nor too easy for the network to solve. When the designed puzzle is complex enough, the network may not converge due to the ambiguity of the task or can easily learn trivial solutions if it is too easy.

Another way to explore the spatial structure of data is by identifying image rotations. Gidaris et al. [44] proposed a new way of learning image representations from unlabeled data by

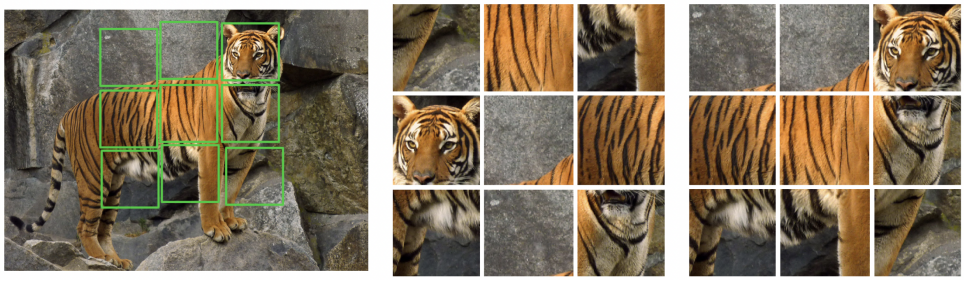

Figure 3.3: Jigsaw Puzzle pretext task. Nine disjoint patches are sampled from the input image. Then, the patches are shuffled randomly (middle image), and each path of the resulting "mosaic" is fed into a neural network. The task is to predict the correct patch ordering to approximate their original space arrangement (right image). Figure reproduced from Noorozi and Favaro [89].

predicting image rotations. The input image is rotated by angles multiples of 90 degrees (0, 90, 180, 270), and then the network outputs which one of the four rotations were applied. The problem formulation implicitly encourages the learned representation to be informative about the object in the image and its rotation. Feng et al. [39] extended the work from Gidaris et al. [44] by jointly predicting image rotations and encouraging rotation-invariant features. The main observation is that rotation transformations might be less applicable for tasks (or images) that are rotation invariant. The latter is achieved by a rotation irrelevant loss, which enforces each rotated image's representation close to their mean latent vector.

Contrastive Learning

Contrastive learning-based pretext tasks draw attention to the whole computer vision field. Over the years, unsupervised or self-supervised methods performed more poorly than their supervised counterpart. However, the game changed since well-designed contrastive learning methods emerged with the adoption of contrastive loss [23, 52, 91]. The core idea is learning to compare or discriminate. For that, the concept of pairs is adopted. A positive pair represents two or more objects representing objects belonging to the same context or expressing some similarity level. We refer the reader the Section 2.2 for a detailed definition of contrastive learning.

DeepInfoMax [58] learns image representations by leveraging the local structure present in an image. The contrastive task is to classify whether a pair of global features (final output of a convolutional encoder) and local features (output of an intermediate layer in the encoder) is from the same image. The model is optimized using contrastive loss by building positive pairs using the local and global representation from the same image while making dissimilar (negative) global and local representations from a random image. Augmented Multiscale Deep InfoMax (AMDIM) [5] enhances the pair construction by sampling positive pairs from two different views (augmentations) of an image.

Contrastive Predictive Coding (CPC) [91] is one of the influential works. Their work is based on predicting the future in the latent space. They treated each image as a timeline, where the top left corner is the past and the bottom right is the future. The input image is divided into a set of overlapped (50%) patches. A neural network encodes each patch to generate a context vector. The future predictions are performed according to the context representations for each patch. While predicting the future information, the CPC is trained to maximize the mutual

information using the InfoNCE loss — mainly formulated on the Noise-Contrastive Estimation loss function (NCE) [49] — between the input image and the context vectors. Here, they use the notion of positive and negative pairs. The positive pairs are sampled according to patches from the same image, while negative ones are sampled from random images on the batch. Latter, Henaff et al. [55] proposed a CPC-V2 by improving CPC for image representation learning.

InstDisc [122] reframes class-level classification as instance-level discrimination: each training sample becomes one label, whose data-augmented views must be recognized against all other training samples. The challenge is extending the loss for many labels (millions, in ImageNet), which is conquered by reformulating the softmax loss. MoCo [52] developed the idea of contrasting using a *momentum contrast*, which substantially improves the portion of negative samples. The authors designed the momentum contrast learning with two encoders (query and key), preventing training instability. They resort to a queue structure (as large as 65,536) to save the newly encoded batches as negative samples. It significantly improves the efficiency of sampling negative pairs. However, MoCo adopts a straightforward strategy to sample the positives: a pair of positive representations come from the same sample without any data augmentation, making the positive pairs easily distinguishable.

Chen et al. [27] investigated how crucial are normalization and negative sampling in contrastive learning. They show that the stop gradient mechanism in BYOL is the most influential component in avoiding representations collapse. They presented a novel pre-training scheme named SimSiam that relies only on neural networks' siamese structure (shared encoder). Their method converges faster than SimCLR, MoCo, and BYOL with smaller batch sizes and minor performance decreases. Their experiments suggested that the siamese structure is crucial for modeling invariances in representation space, which is the heart of representation learning. Tian et al. [108] also explored factors to help SimSiam and BYOL to prevent representation collapse without negative pairs.

The dynamics of learning in these methods and avoiding collapse are not fully understood, although theoretical and empirical studies point to the crucial importance. Based only on the importance of Siamese structure, we have BarlowTwins [123], DINO [18], and VicReg [7]. They all explored variance minimization or relying on the teacher-student training structure.

Another investigation branch in contrastive learning is how to create positive and negative views. Although negative samples might be dispensable in methods like BYOL, and SimSiam, how positive views are generated proved to improve the transfer performance of the learned presentation [17,23]. Other works improve the view generation processes using channels colors and segmentation masks [106], adversarial learning [63,105], causal mechanisms [85], nearest-neighbors in latent space [3,37].

Clustering

Clustering techniques can be incorporated either as a self-supervised loss or for self-labeling. Caron et al. [16] combined the unsupervised clustering and deep neural network to perform a classification task. The method takes augmented unlabeled images as input, and then a convolutional neural network is used to generate a latent representation. The k-means [81] unsupervised clustering algorithm is applied to generate pseudo-labels which are used as ground-truth labels to train the model in an end-to-end way.

Asano et al. [2] used a randomly initialized network to bootstrap a set of image labels. First, a randomly initialized off-the-shelf model generates labels for augmented images. Then, the Sinkhorn-Knopp [33] algorithm is applied to cluster the image representations and produce a new set of labels. New training is performed on this new set of more reliable labels and optimized with cross-entropy loss.

Gansbeke et al. [112] worked towards improving clustering methods by introducing a self-supervised pre-training step. They first used an encoder trained in any self-supervised manner. The encoded representations are clustered by an online clustering module and encourage each representation's nearest neighbors to be close to each other. Finally, the encoder is fine-tuned if the cluster assignment confidence is above a certain threshold. The entropy of the number of samples in each cluster is used to avoid the model assigning all samples into a single cluster.

Table 3.1 overviews each method described in generative and context-based self-supervised sections for general-purpose tasks. We chronologically summarized them according to their sub-category (generative or context-based), main contributions, and publication year.

3.2 Self-Supervised Learning on Medical Tasks

This section discusses how self-supervised learning literature is organized for medical applications. Our inclusion-exclusion criteria rely on works over the period 2017–2021, as this is the period where self-supervised learning appeared in medical imaging analysis. We consider only research papers that either borrow some self-supervised learning scheme from computer vision to solve medical imaging tasks directly or propose a novel self-supervised learning approach leveraging medical knowledge about the target task. We excluded any other works of less relevance.

Suitable pretext tasks are crucial for learning predictive representations, motivating some works to evaluate whether domain-specific might improve self-supervised learning for medical images. Mainly, there are two paths to follow when using self-supervised learning in medical applications. One path is to use the exact pretext task designed for general-purpose computer vision or propose a slightly adapted version of such tasks that best fits the current medical application. The second way is to leverage knowledge about the medical domain — by experience or any domain expert involved — and computer vision to explore a new way to design a custom-built pretext task for the target medical application. In the previous chapter, we prefer to keep the same organization: generative and context-based self-supervision. We restrict our review to only 2D image analysis and classification as the target task. We refer the reader to the following survey on self-supervision applied to 3D imagery [126] and segmentation-only tasks. We refer the reader to the following surveys [28, 99] and the work of Taleb et al. [104].

3.2.1 Generative-Based Self-Supervision

Most generative-based works use pretext tasks created by computer vision literature for natural images. Chen et al. [21] proposed a generative task based on the early works of context encoders [93] and relative patch prediction [35] on magnetic resonance images. Patches of the input image are swapped and must be restored to their proper places. They found the technique advantageous for several downstream tasks, such as fetal MR classification, kidney localization,

Table 3.1: Summary of some selected papers on literature review in self-supervised learning. 'G' stands for generative and 'C' stands for context-based self-supervision.

Ref_{Year}	Method	Category	Contribution	Loss Function
[124]$_{2016}$	Image Colorization	G	Creates a plausible colored version of the input grayscale image	Mean squared error plus color quantization
[93]$_{2016}$	Context Encoder	G	Reconstruct a corrupted part of the image based on its surrounding	Mean squared error and Adversarial [45]
[91]$_{2018}$	CPC	C & G	Learns self-supervised representations by predicting the future in latent space by using autoregressive models	Contrastive learning
[16]$_{2018}$	Deepcluster	C	Leverages k-means algorithms to create pseudo-labels for clustering and a neural network to predict the cluster assigned to each sample	Cross-entropy in assigned clusters
[106]$_{2019}$	CMC	C	Learning by contrasting multi-views of the data. Multi-views include color channels, depth estimation, and semantic segmentation mask estimation	Contrastive learning (multi-vew)
[23]$_{2020}$	SimCLR	C	Contrastive learning of visual representations by projecting and contrasting latent representations using positive and negative views	Contrastive learning
[52]$_{2020}$	MoCo	C	Momentum Network and memory bank to store intermediate feature representations in order to minimize the batch size for negative samples	Contrastive learning with memory bank
[112]$_{2020}$	SCAN	C	Mixing Self-supervised pre-training with online clustering through neighborhood representations aggregation and self-labeling with confidence	Clustering assignment
[48]$_{2020}$	BYOL	C	Improved the contrastive learning setting without the need of negative samples by contrasting two different views representations using ℓ_2 loss, a momentum, critic network and an extra projection head	Mean squared error between positive pairs' representations
[107]$_{2020}$	InfoMin	C & G	Schematic of contrastive representation learning with a learned view generator. An input image is split into two views using an invertible view generator	Contrastive learning
[17]$_{2020}$	SwAV	C	Learning online prototypes for clustering, contrasting clustering assignments for different views as pretext task and multi-crop augmentation policy	Clustering assignment and Sinkhorn-Knopp [33]
[27]$_{2021}$	SimSiam	C	Relies only on the siamese structure of neural network	Mean squared error of ℓ_2-normalized vectors
[123]$_{2021}$	BarlowTwins	C	Relies only on the siamese structure of neural network	Covariance minimization
[7]$_{2021}$	VicReg	C	Relies only on the siamese structure of neural network	Variance-Invariance-Covariance Regularization
[18]$_{2021}$	DINO	C	Teacher-student network and self-distillation	Knowledge distillation to match the teacher's distribution to student's

and brain tumor segmentation. Hu et al. [61] explored the same pretext task task [93] along with DICOM images metadata for ultrasound images. Inspired by SimCLR architecture, they introduced an auxiliary discriminator network that produces a feature vector of the inpainted image to act as input to both classification and projection head. The classification head classifies if the image generated from the context encoder task is real or fake. In contrast, the projection head performs as a conditional classifier that incorporates the DICOM meta-data as weak labels. In eye fundus application, Moris et al. [86] employed the vanilla formulation of context encoders but in a patch-wise manner.

Boyd [14] used GANs to learn representation and produce high-quality samples for digital pathology. They trained a neural network to perform visual field expansion, which progressively increases image generation resolution in curriculum learning. Besides the standard adversarial loss, they used an additional regularization to ensure the Gaussian distribution of the latent representations. Also, in the pathology domain, PathGAN [96] alters the discriminator's goal to estimate the probability of the real data being more realistic than the fake. They borrow two elements from the StyleGAN to allow the generator to produce better feature representation.

Holmberg et al. [59] suggested that designing a practical pretext task for medical domains must accurately extract disease-related features which are typically present in a small part of the medical image. They developed a novel pretext task for ophthalmic disease diagnosis that uses two different image modalities, including optical coherence tomography scans (OCT) and infrared fundus images. Three experienced ophthalmologists have validated their model's predictions. Further, the final performance was assessed on diabetic retinopathy grading using color fundus as a downstream task.

We did not find many papers that perfectly fit our criteria for this section: 2D imaging, purely self-supervised, not multi-tasking, and features classification as target tasks. We refer the reader to the work of Haghighi et al. [50], which focuses on self-supervised for multi-tasking learning, i.e., combining several self-supervised pretext tasks evaluated on several medical datasets.

3.2.2 Context-Based Self-Supervision

Jamaludin et al. [64] pre-trained a Siamese Network with a contrastive loss in which the positive pairs are patches of spinal magnetic resonance images depicting the same vertebrae of a patient across exams, and the negative pairs are corresponding vertebrae in different patients. They found that the scheme improves the prediction of intervertebral disc degeneration grading.

Tajbakhsh et al. [103] exploited several pretext tasks, such as, rotation [44], colorization [77], and GAN-based [45] patch reconstruction. They showed that pretext tasks based on pre-training in the medical domain were more effective than random initialization and transfer learning (ImageNet pre-training) for diabetic retinopathy classification.

Some approaches leverage the available metadata to design pretext tasks or incorporate them to improve representation quality by introducing domain knowledge. Specific medical applications that benefit the metadata are freely available depending on the capture device or annotate extra information by being a common practice.

Li et al. [79] proposed a novel embedding loss function to learn modality and transformation invariant as well as patient similarity features for ophthalmic data. They achieve modality invariance by combining color fundus images with a synthesized fundus fluorescein angiogra-

phy photo of the former image. An additional step is representing transformations invariant by the standard augmentation approaches of the color fundus. Such triplet of photos is assumed to share similar features for the same patient. They consider a triplet of each patient image as a contrasting basis to learn the patient's similar features. Reminiscent of contrastive learning, the features of the same patients are pulled together while features from other patients are pulled apart using the proposed loss function.

Sowrirajan et al. [100] adopted the self-supervised MoCo-based [52] pre-training approach models for chest X-ray classification. They use the original MoCo scheme to pre-train on a large collection o X-ray images, but they initialize the encoder weights with the supervised pre-training on ImageNet to converge faster. They evaluated an external chest X-ray dataset to evaluate the generalization capabilities on tasks from the same domain, which showed a promising approach by increasing the mean performance. Vu et al. [116] extended the previous work by introducing an augmentation strategy that leverages the patient metadata to sample the positive views. MoCo pre-training appears to be widely used in the medical field, bringing superior performance to other medical applications compared to their supervised counterpart for COVID diagnosis [101, 116], and pleural effusion classification [25].

Ciga et al. [31] investigated the SimCLR contrastive pre-training for digital histopathology in several classification and segmentation tasks. They find that combining multiple multi-organ datasets with different types of staining and resolution improves the quality of the learned features. In addition, contrastive pre-training using only in-domain images achieved better performance at the target tasks (breast cancer and tissue classification) than the models trained in a supervised manner on ImageNet. A range of solutions were proposed for for histopathology images to incorporate self-supervised representation learning, such as clustering assignment regression [87], Transformer-based pre-training [119], exploring spatial proximity [1], and inter/intra-class variance [78].

Azizi et al. [4] investigated two medical tasks: skin-lesion analysis on a private dataset of $> 450,000$ teledermatology clinical images and X-rays on the publicly available CheXpert dataset. Contrasting SimCLR pre-training to two strong supervised pre-training baselines, they find it advantageous for the skin-lesion task and similar for the X-rays task. They introduced the Multi-Instance Contrastive Learning (MICLe), which is based on SimCLR [23] with minor modification. The main idea behind MICLe is to leverage the metadata information from the same patient as the foundation for contrastive learning. They encourage samples from the same patient to be close in representation space while setting apart representation from negative pairs. We summarized some of the self-supervised learning applications in medical domain in Table 3.2.

Table 3.2: Selected works we covered in our review of self-supervised learning for medical applications. We highlight their contributions, pretext tasks and the target tasks.

Ref_{Year}	Authors	Pretext Task	Target Tasks
[64]$_{2017}$	Jamaludin et al.	Uses a siamese network to learn embeddings by comparing image pairs' representations A second pretext task used is predicting vertebral body levels.	Disc degeneration grading
[87]$_{2019}$	Muhammad et al	Reconstruction error and clustering centroid regression	Cholangiocarcinoma (liver cancer) subtyping
[1]$_{2020}$	Abbet et al.	Uses contrastive learning by sampling positive pairs from overlapped image patches and an additional relative entropy term in respect to the neighbouring in latent space	n/a
[79]$_{2020}$	Li et al.	Patient feature-based sofmax embedding	Diabetic retinopathy detection, age-related macular degeneration classification, and pathological myopia classification
[100]$_{2021}$	Sowrirajan et al.	MoCo-based pre-training	Tuberculosis detection and pleural effusion classification
[101]$_{2021}$	Sriram et al.	MoCo-based pre-training	COVID-19 patient prognosis
[116]$_{2021}$	Vu et al.	Uses available patient metadata to improve pairs sampling for contrastive learning. They also studied several ways to construct the negative pairs.	Classification of pleural effusion in chest X-ray images
[31]$_{2021}$	Ciga et al.	SimCLR-based pre-training	Breast cancer, prostate cancer, lymph node, colorectal cancer tissue classification
[4]$_{2021}$	Azizi et al.	Proposes a contrastive learning scheme based on SimCLR leveraging available matadata and multiple patient conditions for contrastive learning	Classification of chest X-ray and dermatologic images

Chapter 4

Methodology

This chapter covers the details of our methodology. We describe each step in the next sections, focusing on reproducible research, facilitating work reproduction, and further extension. In Section 4.1, we overview our pipeline and the investigated methods (MoCo [26,52], InfoMin [107], SimCLR [23], BYOL [48], and SwAV [17]) on in- and out-distribution scenarios. We highlight that part of this chapter is available as a pre-print on ArXiV platform [19]. In Subsection 4.1, we present our investigated pipelines and how we conduct the experimental protocol describing how we organized all experiments.

One of our work's main objectives is to provide a fair systematic evaluation of self-supervised models. We strictly describe and follow our designed pipeline during the experimental design to guarantee reproducible research. Although we focus only on image data, the pipeline is not restricted to only images; but can easily adapt to several data inputs such as text, audio, or video applications. Figure 4.1 shows our pipeline. It is composed of four main stages:

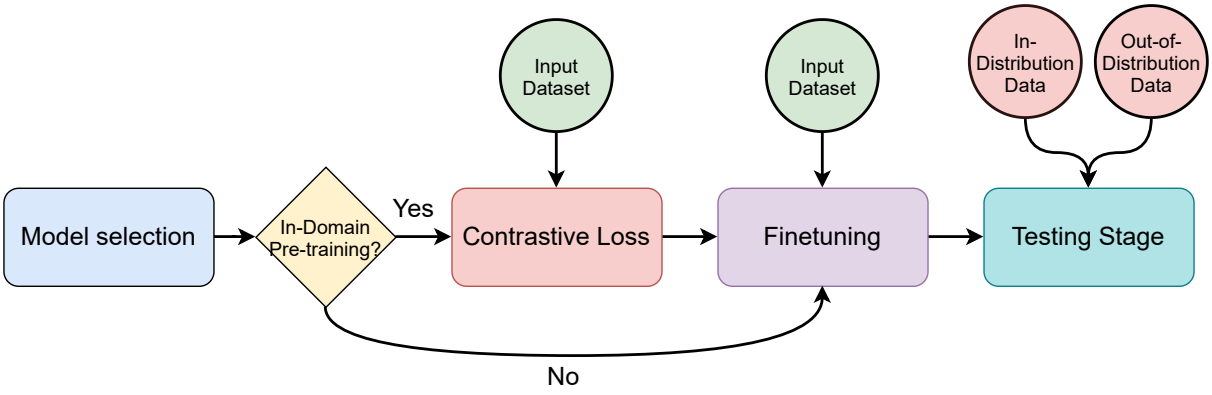

Figure 4.1: Proposed evaluation pipeline.

1. **Model selection:** It compromises in selecting which neural network is being used as the main feature extractor. Another reasonable choice is if the chosen network will be pre-trained or not. Pre-training can be either supervised or self-supervised based. This step is similar to the standard transfer learning protocol: choose a pre-trained model as a starting point for further experiments or fine-tuning.

2. **Contrastive loss**: Optional stage. We perform an additional contrastive learning pre-training after the model selection, independently if the chosen model in the previous stage

was once pre-trained or not. The additional contrastive training uses the same data as in the fine-tuning stage. It aims to evaluate if an additional pre-training using contrastive loss is advantageous using in-domain data. We adopted the contrastive loss (among various pretext tasks available in the literature) for two main reasons: 1) The vast majority of pretext tasks are hard to adapt to a supervised scenario. A straightforward solution is to train the model in a multi-task fashion: combining both self-supervised loss and supervised loss (e.g., cross-entropy) as final loss; 2) There is a supervised version [70]. Introducing the label information in contrastive loss is made implicitly: they help to contrast correctly the positive and negative pairs, which leads to better representation for the target downstream task [70], which is far from the traditional (cross-entropy) label classification. Evaluating both self-supervised and supervised versions of contrastive loss allows us to understand the impact of having labeled data in this stage.

3. **Fine-tuning**: The step where we fine-tune the given model to the target domain and task using domain-specific data. Here, we need to adjust the hyperparameters according to the input model since many disagree on the best set of parameters for training. Our experiments show that such sensibility to hyperparameter selection also depends on the number of samples in each dataset.

4. **Testing**: Machine learning models often need to generalize from training data to new environments. The standard procedure to measure generalization is to evaluate a model on a single test set drawn from the same distribution as the training set. It is hard or impossible to train a model on precisely the distribution it will be applied to in many scenarios. Hence a model will inevitably encounter out-of-distribution data on which its performance could vary compared to in-distribution performance. Especially in medical applications, such distribution shifts can occur depending on the capture device or data-driven biases, such as underrepresented population [82]. After fine-tuning stage, we perform a hold-out test stage, which comprises both in- and out-distribution scenarios for each dataset used in training. We carefully created two test sets for each dataset: in- and out-distribution sets. Such protocol aims to evaluate the robustness of the trained models under distribution shits, which is close to a realistic use case, especially for medical applications. We highlight our contribution to systematically evaluating carefully crafted in- and out-of-distribution scenarios for medical and general case applications. Our work is the first in the medical literature to extensively assess both supervised and self-supervised pre-training in several medical applications for in- and out-of-distribution. Most close to our work is Miller et al. [84] which performed a similar analysis on standard computer vision datasets. In the medical domain, Hosseinzadeh et al. [59] and Truong et al. [110] evaluate the transferability performance of self-supervised methods to medical applications but evaluate the performance in distribution-shifted scenarios is omitted.

We stand out our work from the ones available in the literature in Table 4.1 by highlighting the contributions of our experimental design.

Work$_{year}$	#Evaluated Tasks	Applications	#Evaluated Methods	Out-of-distribution Evaluation	Low-data Evaluation
Azizi et al. [4]$_{2021}$	2	Medical	2	No	Yes
Miller et al. [84]$_{2021}$	15	Natural & Medical	12	Yes	No
Hosseinzadeh et al. [60]$_{2021}$	7	Medical	15	No	No
Truong et al. [110]$_{2021}$	4	Medical	5	No	Yes
Ours$_{2022}$	7	Natural & Medical	6	Yes	Yes

Table 4.1: Overview of works that evaluate self-supervised versus supervised pre-training.

4.1 Pipelines

We organized our pipelines to investigate two main fronts: **skin lesion** and **general-purpose case**. The former has three rounds of experiments, and the latter has only two rounds. We detail how we conducted the experiments for each front in Sections 4.1.1 and 4.1.2. First, we started our analysis with skin lesion classification as the primary application due the candidate's research group has been widely studying skin lesion analysis since early 2014 [40]. The group is at the forefront of such research worldwide, responsible for groundbreaking results associated with skin lesion analysis. Since self-supervised learning has made enormous progress in 2020 and proved to be advantageous in several downstream tasks compared to the traditional supervised learning [18,23,46], such behavior inspired us to investigate if self-supervised pre-trained models would bring superior performance for skin lesion analysis. We decided to extend a subset of our experimental setup to other medical applications and for the general-purpose case.

At the beginning of our research (January 2021), we chose the five self-supervision scheme candidates by selecting techniques with pre-trained weights (ResNet-50 $1\times$) made available by the original authors and ranked them on the top-1 accuracy on ImageNet. We selected the most recently published. We are aware that the top-5 ranking has changed, but we did not include those new models in our analysis due to time constraints.

We explore five self-supervised approaches for comparison against a supervised baseline. We follow standard trends in self-supervised and adopt ResNet-50 ($1\times$) [53] as the main backbone for all experiments due to the wide adoption of such network architecture in experimental sections for self-supervised learning. Hence, it is common to release at least the trained model for ResNet-50 ($1\times$).

4.1.1 Skin Lesion Case

As mentioned earlier, we explore three rounds of experiments in the skin lesion case. In the first round, we attempt a few combinations of hyperparameters for each self-supervision scheme. We purposefully optimize the baseline pipeline more thoroughly to make it challenging. The exact search space appears in Table 5.2.

First, we compare the baseline pipeline with the typical self-supervision pipeline to establish whether self-supervision is advantageous. In addition, we select a self-supervision scheme among five candidates (BYOL, InfoMin, MoCo, SimCLR, and SwAV) to perform the remainder of the experiments. Selecting the most promising scheme at this stage is necessary for managing the number of experiments, as the next round of experiments will be exhaustive and,

thus, expensive.

According to the model's performance, the second round of experiments investigates performing in-domain pre-training using the best self-supervised scheme. We hypothesize that in-domain pre-training might boost the model performance on the target task by providing a sound feature adaptation on fine-tuning.

The third round consists of a systematic evaluation of all pipelines under three data regimens: full training data with 100% of the samples, and low-training data with 10% and 1% of the samples. The latter intends to simulate the frequent scenario of insufficient training data on medical images. Next, we intend to evaluate the generalization capabilities of all trained models on one in-distribution dataset and four out-of-distribution datasets.

We evaluate four alternative pipelines (Figure 4.2), which vary in the pre-training and fine-tuning of the model. All pipelines finish with a fine-tuning (FT) on the train split. The traditional supervised pipeline (SUP $\rightarrow$ FT) is pre-trained using classical, supervised learning on ImageNet. All self-supervised pipelines (SSL $\rightarrow$ *) are pre-trained using self-supervision (without class annotations) on ImageNet. The SSL $\rightarrow$ * $\rightarrow$ FT pipelines have an additional, intermediate pre-training step on the train split using supervised (SCL) or unsupervised (UCL) contrastive loss (Section 4.1.1).

For each combination of pipeline and hyperparameter, we measure their performance on the validation split five times, reflecting different random initializations for the training procedures, and, on the low-data experiments, we sampled different random training subsets. We perform five replicates on the test split for each combination, resulting in 25 measurements for each pipeline. The hyperparameters, fixed and variable (factors and levels) evaluated for all pipelines are detailed in the following subsections.

SUP $\rightarrow$ FT baseline pipeline

In this pipeline, we start from a ResNet-50 model pre-trained with a classical supervised loss on ImageNet and perform a fine-tuning, using a supervised loss on a skin lesion dataset. A simple linear classifier is trained on top of an encoder network jointly in a supervised way using cross-entropy as the loss function. The weights from MLP are initialized randomly.

SSL $\rightarrow$ FT pipeline

We start from a ResNet-50 model pre-trained with *self-supervised* losses on ImageNet and then perform a fine-tuning using a supervised loss on the same skin lesion datasets that will be used in fine-tuning. We start from the best publicly available checkpoints for each self-supervised scheme we evaluate. For each model, we add a binary random-initialized linear layer to the network's output, feeding to a cross-entropy loss function.

SSL $\rightarrow$ UCL/SCL $\rightarrow$ FT pipelines

We choose one of the five evaluated self-supervised methods to perform an additional pre-training stage, i.e., continue the original pre-training task, using its original formulation, and then perform the exact fine-tuning protocol as detailed in the previous sections. We chose

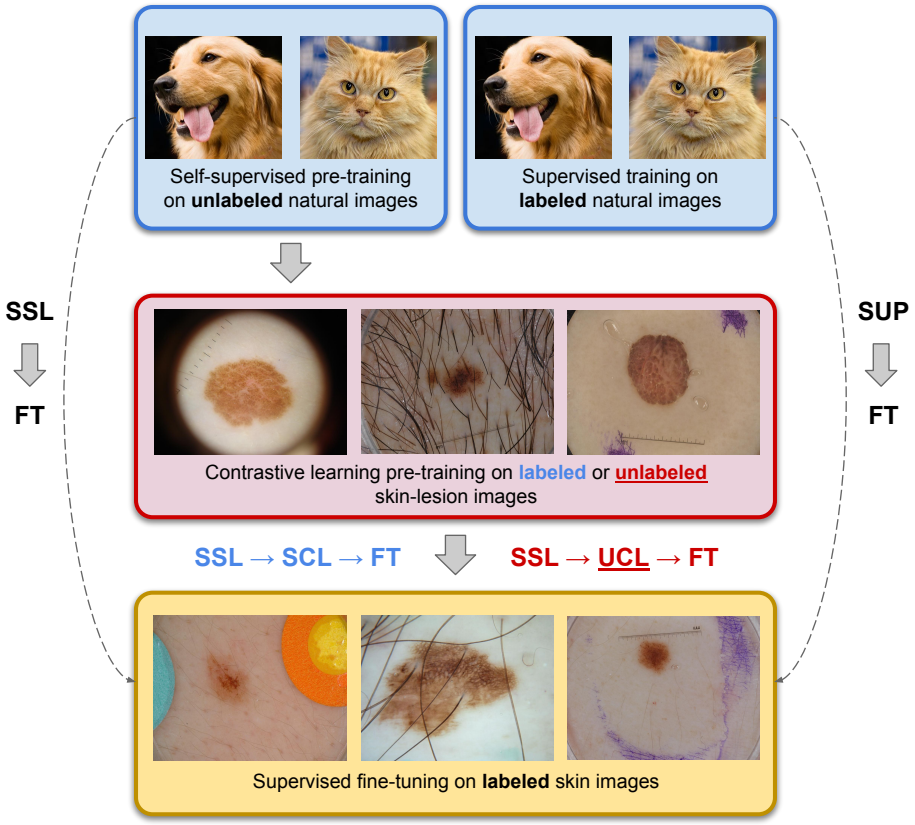

Figure 4.2: Overview of our evaluated pipelines. In SSL → FT scheme we contrast the result of five fine-tuned SSL ImageNet pre-trained models on skin lesion dataset (see Section 5.1.1) with the supervised counterpart. The SSL → SCL → FT pipeline differs from SSL → UCL → FT according to the employed contrastive loss. They both go through a pre-training stage — which can be supervised (SCL) or unsupervised (UCL) — and then performing a supervised fine-tuning. We test all trained models on a hold out test set and five out-of-distribution datasets to assess the generalization performance. Figure inspired from Azizi et al. [4].

SimCLR because of its relatively simple implementation cost compared to the other approaches and good performance presented in the fine-tuning only experiments (see Section 5.1).

Originally, SimCLR pre-training was performed using the self-supervised contrastive loss function. However, we investigated to incorporate and evaluate its respective supervised contrastive loss [70] version. As such, we created two experimental setups:

Unsupervised Contrastive Learning (UCL): The model weights are initialized using the best encoder checkpoint publicity available for ResNet-50 (1×) on SimCLR. Then, the fine-grained representations are **refined** under a self-supervised contrastive framework using the isic2019 training set. This is the most straightforward approach to try. In many cases, a large collection of unlabeled data is available, but those are unused due to the lacking of annotation. Although we leave as unexplored the approach of combining multiple datasets or unlabeled samples as extra data during model optimization, that is a viable approach we intend to explore in future work. We indicate this pre-training initialized using self-supervised ImageNet weights as **SSL → UCL → FT** in the following sections. We remind the reader of the self-supervised contrastive loss in Equation 2.1.

Supervised Contrastive Learning (SCL): Similarly, as detailed in self-supervised pre-

training, but uses the supervised contrastive loss (equation 4.1) instead of the self-supervised one. This approach may be desirable when the label information is available. This pipeline's main benefit over the self-supervised version is improving sample selection since it plays an important role in model optimization. This allows contrasting negative pairs or samples correctly (rather than at random, as in the self-supervised version) and positive pair representation agreement. For this reason, we choose to assess if the supervised contrastive loss benefits the final performance. We hypothesize that the model may learn sharper decision boundaries once the label information is known *a priori*. Besides, we explore having balanced class distributions in the mini-batches. We indicate this pre-training initialized using self-supervised ImageNet weights as **SSL** $\rightarrow$ **SCL** $\rightarrow$ **FT** in the following sections.

$$\mathcal{L}_{SCL} = \frac{-1}{2N} \sum_{i=1}^{2N} \frac{1}{|Z_i^+|} \sum_{z^+ \in Z_i^+} \log \frac{\exp(z_i \cdot z^+)/\tau}{\sum_{k \neq i}^{2N} \exp(z_i \cdot z_k)/\tau}, \tag{4.1}$$

where $z_j = f_\theta(x_j)$ is a vector representation of the input image x_j parameterized by neural network f_θ with parameters θ, z_j^+ and z_j^- are, respectively, positive and negative samples, $sim(\cdot, \cdot)$ is any function that computes the similarity between two vectors, $\tau > 0$ is a scalar temperature hyperparameter. N is the batch size. Z_i^+ is the number of positive samples for the given label.

4.1.2 The General Case & Other Medical Applications

This section highlights our contribution to extending the previous experimental design to various other classification problems. We keep the same core objective to evaluate the performance of fine-tuned pre-trained models on both in- and out-distribution test sets and when low data are available for each application. We decided to expand a subset of our previous analysis to investigate if our observations in skin lesion analysis translate to other medical applications or the general-purpose computer vision case.

There are plenty of medical and general case applications we could evaluate. As we indented to assess models in scenarios close to real-world distribution shifts, we need to design such scenarios. We choose applications that either make available metadata information that lets us make accurate data splits to mimic distribution shifts (e.g., in the training set only have cat images in domestic environments, but cats appear in wood environments in the test set) or largely studied applications with two (or more) data sources aiming at the same target classification task. Finally, We use two datasets for the general-purpose case applications and eight datasets for medical applications, covering four different classification tasks (breast cancer, histopathologic tissue, brain tumor) and imagery type. We describe each dataset in Section 5.2.1 about the number of samples and classes and the data source.

We intend to evaluate the same five self-supervised models as mentioned in the skin lesion case (Subsection 4.1.1) and contrast their results against a supervised baseline. Figure 4.3 depicts all evaluated pipelines. We investigate only two alternative pipelines: supervised (SUP $\rightarrow$ FT) and self-supervised (SSL $\rightarrow$ FT) fine-tuning. We discarded all pipelines with the additional pre-training for two reasons: 1) It would increment *a lot* our experimental design in terms of running time. To find the parameters for the pre-training requires running the whole protocol of employing self-supervised pre-training followed by the standard fine-tuning; thus,

expensive in terms of computational resources; 2) We struggled to find an appropriate hyper-parameter set for all evaluated models in fine-tuning that works well on all (or majority) target tasks. Unlike skin lesion analysis, where the initial investigated parameters seemed to be good candidates for fine-tuning, we experienced a large variance in performance depending on the training set and parameter combination. Ideally, we should perform an exhaustive grid search of hyperparameters for each dataset (6), data split (5), and pre-training scheme (baseline plus five self-supervised) for all training percentages (3). However, such a protocol is costly and requires high-scale computational resources to finish one entire batch of experiments. Instead, we only perform the grid search for each pre-training scheme using the first split for each percentage. The best hyperparameter combination is replicated to other splits to diminish the number of experiments.

We also perform the systematic evaluation of the two pipelines under three data regimens: full training (100%), and low-training data with 10% and 1% of the samples. Finally, we pose both models in challenging test scenarios comprising in- and out-of-distribution datasets to measure the model's generalization capabilities. A common practice in the literature to introduce distribution shifts in test sets is by modifying the original content in pixel space by adding noise, perturbations, or corruptions [56]. However, such a procedure is highly artificial and fails to introduce realistic distribution shifts in medical scenarios.

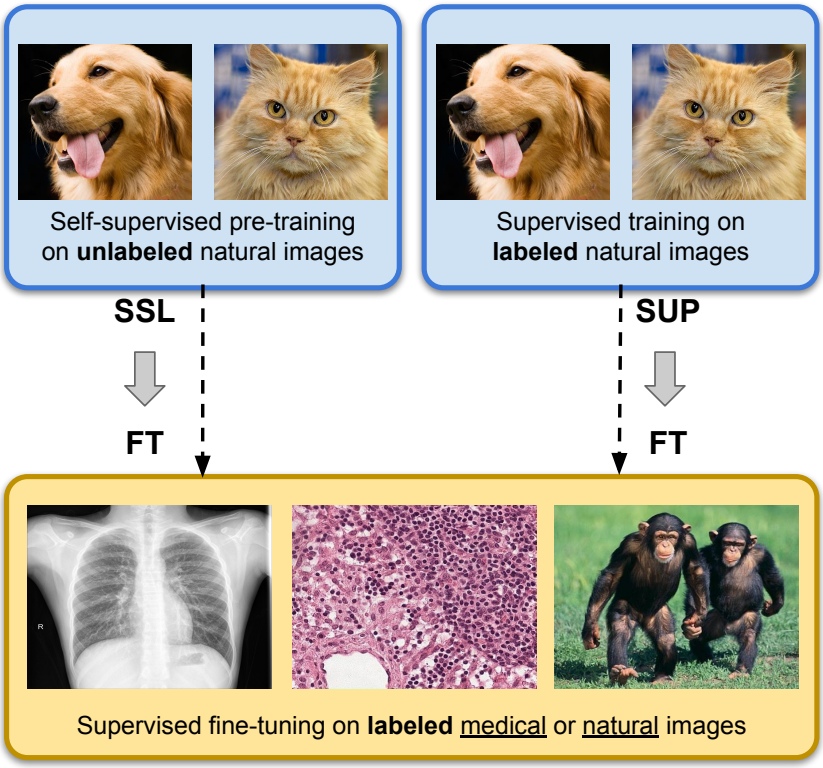

Figure 4.3: Overview of our evaluated pipelines for both other medical and natural applications. In SSL $\rightarrow$ FT scheme we contrast the result of five fine-tuned SSL ImageNet pre-trained models on medical and general purpose (see Section 5.2.1) with the supervised counterpart (SUP $\rightarrow$ FT). We test all trained models on a hold-out test set and five out-of-distribution datasets to assess the generalization performance.

SUP $\rightarrow$ FT & SSL $\rightarrow$ FT

In both pipelines, we start from a ResNet-50 model pre-trained with a classical supervised (SUP) or self-supervised (SSL) loss on ImageNet and perform a fine-tuning, using a supervised loss on the proper training set, either medical or general-purpose. A random-initialized simple linear classifier is trained on top of an encoder network jointly in a supervised way using cross-entropy as the loss function.

Chapter 5

Results

This chapter shows all results regarding each experimental design explained in Section 4.1. We cover four medical (skin lesion, breast cancer, brain tumor, and cancer tissue) and three general-purpose image classification tasks. We follow the same presentation as in the previous chapter and first describe the results for the skin lesion scenario, and then we explore and discuss the results considering the other medical and natural applications. We highlight our contributions, especially in organizing and designing out-of-distribution test scenarios to evaluate model performance to distribution shits.

5.1 Skin Lesion Case

5.1.1 Evaluation Metrics & Datasets

Following the International Skin Imaging Collaboration (ISIC) 2020 Challenge [98], our task is melanoma *versus* benign lesion classification. We perform an end-to-end fine-tune with a single linear layer on the top of the encoder. As the skin lesion datasets are commonly unbalanced in terms of the number of samples — with a majority being benign — we compare the achieved area under the ROC curve (AUC), always testing the final model with both in- and out-of-distribution samples. We evaluate our experiments in five high-quality, publicly available datasets (Table 5.1).

We also pose supervised and self-supervised models in scenarios where the test set's data distribution differs from training ("cross-dataset") to mitigate model bias [42]. So, the resulting array of testing sets comprises similar images ("in-distribution") and distribution-shifted images ("out-of-distribution") to measure our model's generalization ability. To test in cross-dataset scenario, we use the derm7pt [68] to create clinical (derm7pt-clinic) and dermoscopic (derm7pt-derm) scenarios; and pad-ufes-20 [92]. Next, we give a brief description of each dataset:

isic19 [32]: It is composed of only dermoscopic skin images. These images are captured with a device called a dermatoscope that normalizes the light influence on the lesion, allowing it to capture sharper details. Specialists diagnose melanoma with a technique called *dermoscopy*, which analyzes the dermoscopic attributes present in the lesion. These attributes are only visible in dermoscopic images. We performed all training ($14,805$ samples) and validation ($1,931$ samples) in splits of the isic19 dataset. We removed basal

and squamous cell carcinomas from all datasets, leaving melanoma as the only malignant class.

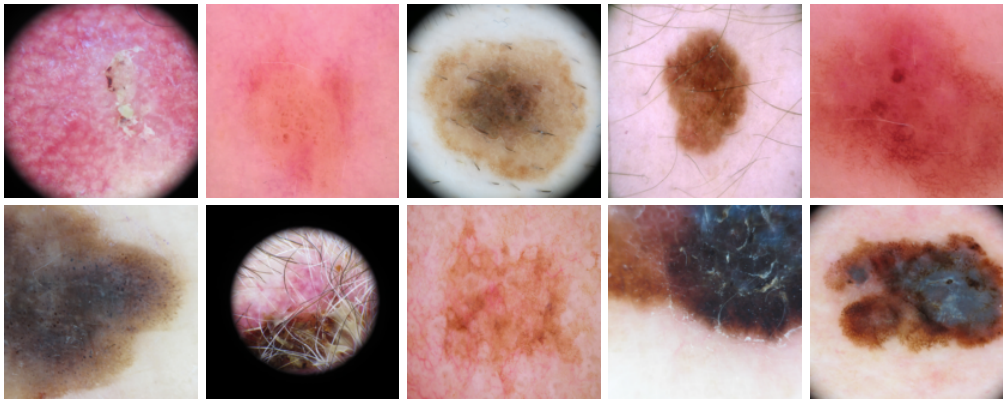

Figure 5.1: Samples from isic19 dataset. The first row shows benign samples, and the second row the malignant ones.

isic20 [98]: In-distribution dermoscopic dataset used only in testing stage. We take a random subset of the original training set, following the 2 (benign) : 1 (malign) class ratio. We remove all duplicates between isic19 and isic20 to create a fair scenario and prevent contamination between train and test sets. The final slit contains $1,743$ images (581 maligns *versus* $1,162$ benign).

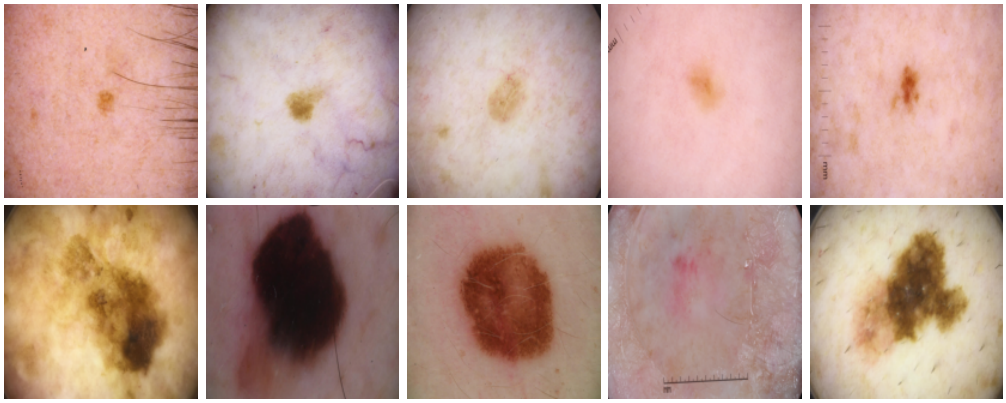

Figure 5.2: Samples from isic20 dataset. The first row shows benign samples, and the second row the malignant ones.

derm7pt-derm [68]: Out-of-distribution dermoscopic dataset. Although derm7pt-derm is composed of only dermoscopic images (as in isic19), we label this dataset as an out-of-distribution case due to differences in the data source. The final split contains 827 images (252 maligns *versus* 620 benign).

derm7pt-clinic [68]: Out-of-distribution clinical dataset. Clinical images differ from dermoscopic according to their capturing device. Clinical images are captured using standard cameras instead of a dermatoscope. Such detail usually alters the image-data distribution. This way, we evaluated the model's generalization capabilities beyond ISIC images. The final split contains 839 images (248 maligns *versus* 591 benign).

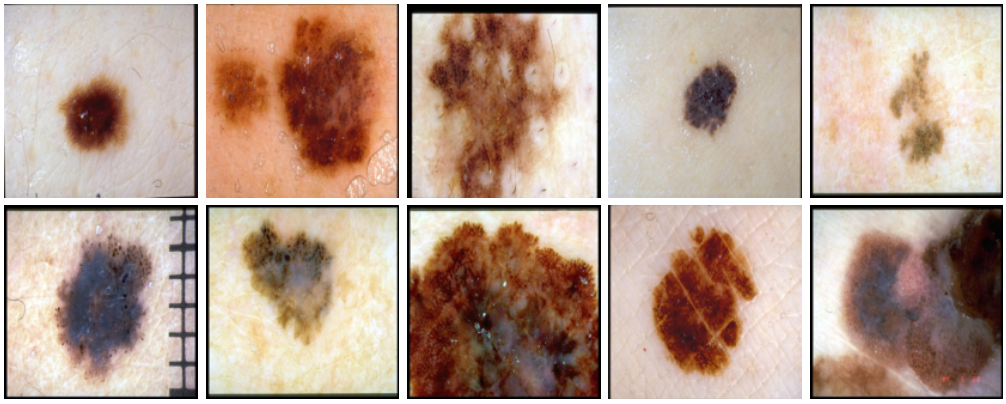

Figure 5.3: Samples from derm7pt-derm dataset. The first row shows benign samples, and the second row the malignant ones.

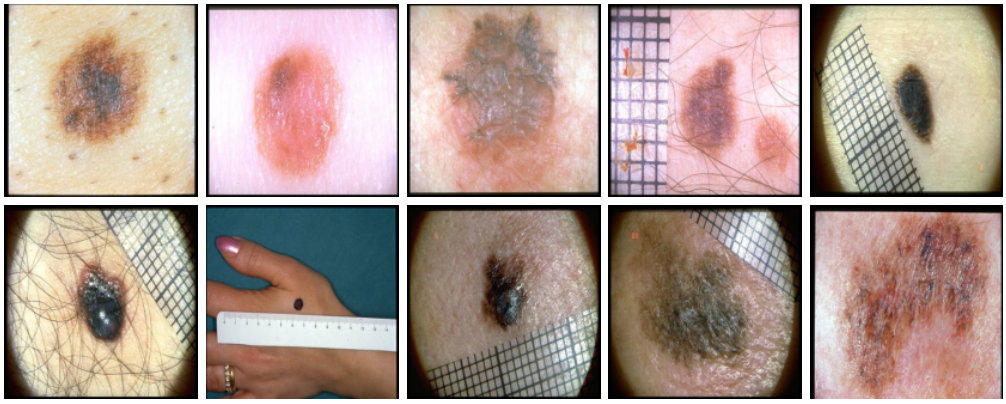

Figure 5.4: Samples from derm7pt-clinic dataset. The first row shows benign samples, and the second row the malignant ones.

pad-ufes-20 [92]: Out-of-distribution clinical dataset. It is a skin lesion dataset collected along with the Dermatological and Surgical Assistance Program at the Federal University of Espírito Santo (Brazil). It comprises clinical images and several patient metadata, such as skin type and lesion location. We removed all non-related classes and introduced a new out-of-distribution malignant class. The final split contains $1,261$ images (52 maligns *versus* $1,209$ benign).

Figure 5.5: Samples from pad-ufes-20 dataset. The first row shows benign samples, and the second row the malignant ones.

Dataset (split†)	Size	Mel.	Lesion diagnoses	Other information
isic19 [32] (train)	14 805	3121	Melanoma *vs.* actinic keratosis, benign keratosis, dermatofibroma, melanocytic nevus, vascular lesion	Dermoscopic images.
isic19 (validation)	1 931	224	Idem	Dermoscopic images, in-distribution.
isic19 (test)	3 863	396	Idem	Idem.
isic20 [98]	1 743	581	Melanoma *vs.* actinic keratosis, benign keratosis, lentigo, melanocytic nevus, unknown (benign)	Dermoscopic images, out-of-distribution, additional unknown diagnosis.
derm7pt–derm [68]	872	252	Melanoma *vs.* melanocytic nevus, seborrhoeic keratosis	Dermoscopic images, out-of-distribution.
derm7pt–clinic [68]	839	248	Melanoma *vs.* melanocytic nevus, seborrhoeic keratosis	Clinical images, out-of-distribution.
pad-ufes-20 [92]	1 261	52	Melanoma *vs.* actinic keratosis, Bowen's disease, nevus, seborrheic keratosis	Clinical images, out-of-distribution, additional Bowen's disease diagnosis.

Table 5.1: Description of the datasets used in skin lesion scenario. Mel.: number of melanomas. †Split used for test if omitted.

5.1.2 Pipeline's Hyperparameters

SUP → FT baseline

We start from a ResNet-50 model pre-trained with a classical supervised loss on ImageNet and perform a fine-tuning, using a supervised loss on the isic19 [32] training split. A simple linear classifier is trained on top of an encoder network jointly in a supervised way using cross-entropy as the loss function. The weights from MLP are initialized randomly.

We strive to make the baseline challenging, by performing, on the isic19 validation split, a thorough grid search comprising batch size $(32, 128, 512)$, balanced batches (yes or no), starting learning rate $(0.1, 0.05, 0.005, 0.009, 0.0001)$, and learning rate scheduler (plateau, cosine). The optimizer is the Stochastic Gradient Descent (SGD) with a momentum of 0.9 and weight decay of 0.001. The plateau scheduler has the patience of 10 epochs and a reduction factor of 10. The fine-tuning last for 100 epochs with early stopping with the patience of 22 epochs, monitored on the validation loss. Both schedulers have a minimum learning rate of 10^{-5}.

SSL → FT

We start from a ResNet-50 model pre-trained with *self-supervised* losses on ImageNet and then perform a fine-tuning using a supervised loss on the isic19 training split. We start from the best publicly available checkpoints for each self-supervised scheme we evaluate. For each model, we add a binary random-initialized linear layer to the network's output, feeding to a cross-entropy loss function.

SSL → UCL/SCL → FT pipelines

Following the original SimCLR implementation, two fully-connected layers are used to project the ResNet representations to 128-dimensional embeddings, used in a contrastive loss. We use Adam [72] as the main optimizer and cosine decay over the epochs. For data augmentation, we performed heavy image augmentation as in SimCLR, composed of color jitter, horizontal and vertical flips, random resized crop, and grayscale with a probability of 0.2. Unlike the original set of proposed augmentation in SimCLR, we discarded the Gaussian blur because we believe it can cause a possible loss of variation and characteristics between regions, harming the final classification. In our pre-training experiments, the images are resized to 224×224.

The factors evaluated in this pipeline comprise items (a) to (e) from Table 5.2. The temperature factor needs to be adjusted according to each specific problem. We select the values $\{0.1, 0.5, 1.0\}$ because these were originally reported and evaluated by the original SimCLR. Longer training also provides more information about the negative samples, and larger batch sizes tend to produce better representations, boosting the final performance in downstream tasks [23, 70, 106]. To verify whether the result holds for skin lesion analysis, we trained all models for 200 epochs and chose the checkpoints for the epochs 50 and 200 for fine-tuning. We use fixed learning rate of 0.001 for the pre-training phase. Once pre-training is finished, we run a fine-tuning procedure described in Section 4.1.1 with the best learning rate that leads to the best results.

Item	Factor	Level
a)	Pretraining contrastive loss	supervised *versus* self-supervised
b)	Pretraining batch size	$\{80, 512\}$
c)	Balanced batches	absent *versus* present
d)	Temperature scale	$\{0.1, 0.5, 1.0\}$
e)	Pretraining epochs	$\{50, 200\}$
f)	Learning rate fine-tuning	$\{10^{-2}, 10^{-3}\}$

Table 5.2: Factors and levels of our experimental design for skin lesion classification. Items a) to e) regard from contrastive learning pre-training (SSL → * → FT), whereas f) corresponds for the learning rate in all fine-tuning experiments (except for the baseline).

Final fine-tuning for all pipelines, Testing

We fine-tune every model using an SGD optimizer with a momentum of 0.9 and weight decay of 0.001, and a plateau scheduler with the patience of 10 epochs and reduction factor of 10. The fine-tuning last for 100 epochs with early stopping with the patience of 22 epochs, monitored on the validation loss. Notice that for the baseline pipeline, we performed additional optimizations (Section 4.1.1).

We resize input images to 299×299. Except for SimCLR, which uses raw inputs, we z-normalize the inputs per channel with statistics from ImageNet. We augment training data with random horizontal and vertical flips, random resized crops containing from 75 to 100% of the original image, random rotations from -45 to $45°$, and random hue change from -20 to 20%.

We apply the same augmentations on train and validation. We also use test-time augmentations [111], averaging the predictions over 50 augmented versions of each test image [95].

We perform all searches on the validation split to avoid using privileged test information on this step [111]. To estimate the statistical variability of those experiments, we perform five replicates for every experiment, reflecting different random initializations for the training procedures (optimizer, scheduler, and augmentations).

5.1.3 Self-Supervision Schemes *versus* Baseline Comparison

As explained in Section 4.1, we organized our extensive experimental design in two rounds, corresponding to this and the next two following subsections. In the second subsection, we analyze the second round of experiments in the specific scenario of low training data.

In this first round of experiments, we compared the baseline pipeline (SUP → FT) to the basic self-supervision pipeline (SSL → FT) with five self-supervision schemes (BYOL, InfoMin, MoCo, SimCLR, and SwAV). We optimized the baseline and the self-supervised pipelines as explained in Section 4.1.1. Finally, we fine-tuned both models for the target task as explained in Section 5.1.2.

The results (Table 5.3) show that, despite having no access to the labels during the pre-training and being less thoroughly optimized during the final fine-tuning, the models with self-supervised pre-training are very competitive. Indeed, two of the pipelines (SimCLR and SwAV) had averages above the ones in the baseline. SimCLR, SwAV, and BYOL benefit from higher learning rates than the hyperoptimized supervised counterpart.

This first round of experiments intended to validate the applicability of self-supervised learning and select one self-supervised scheme for the expensive round of systematic evaluations in the next round. Thus, it comes with the caveat that both optimization and evaluation were conducted in the isic19 validation set. The second round of experiments will evaluate the ability of the pipelines to generalize performance in the rigorous setting of a hold-out test set.

Method	AUC (%)	Hyperparameters			
		learning rate	batch size	batches	scheduler
Supervised (baseline)	94.8 ± 0.6	0.009	128	balanced	plateau
SimCLR [23]	**95.6 ± 0.3**	0.01	32	unbalanced	plateau
SwAV [17]	95.3 ± 0.6	0.01	32	unbalanced	plateau
BYOL [48]	94.6 ± 0.5	0.01	32	unbalanced	plateau
InfoMin [107]	94.4 ± 0.5	0.001	32	unbalanced	plateau
MoCo [52]	93.9 ± 0.7	0.001	32	unbalanced	plateau

Table 5.3: The best results for the first round of experiments, comparing the supervised SUP → FT baseline to the basic SSL → FT pipeline with five SSL schemes. The metric is the AUC on the isic19 validation split. Despite the baseline using label information on pre-training, and being more thoroughly optimized, self-supervision pre-training is still very competitive with it.

5.1.4 Systematic Evaluation of Pipelines

In the second round of experiments, we performed a systematic evaluation of the baseline pipeline, pre-trained with supervision (SUP $\rightarrow$ FT) against the three pipelines pre-trained with additional contrastive learning loss self-supervision (SSL $\rightarrow$ FT, SSL $\rightarrow$ UCL $\rightarrow$ FT, and SSL $\rightarrow$ SCL $\rightarrow$ FT). In this round, we only evaluated SimCLR as the self-supervision scheme for several reasons: it showed the best performance in the preliminary experiments, it allows introducing annotation information easily with a supervised contrastive loss, it has one hyperparameter less than SwAV to optimize (number of clusters), and the ablation studies in the original papers helped to decide on a range of reasonable values for the temperature value.

As explained in Section 4.1.1, this round of experiments simulates a realistic machine-learning protocol, in which first we optimize the hyperparameters for each pipeline on the isic19 validation split, then evaluate the performance on a hold-out test set. The test set may be the in-distribution isic19 test split, or the out-of-distribution isic20, derm7pt-derm, derm7pt-clinic, and pad-ufes-20. Those cross-dataset evaluations are critical to evaluate how well the pipelines generalize to different classes, image acquisition techniques, or even to subtle dataset variations across institutions.

The results appear in the topmost plot of Figure 5.6, where each boxplot shows the distribution of 25 individual measurements (small black dots), corresponding to the best five non-unique hyperparameterizations, with five replicates for each of them. The boxplots show, as usual, the three quartiles (box), and the range of the data (whiskers) up to $1.5\times$ the interquartile range (samples outside that range are plotted individually as "outliers"). The large red dots show the means for each experiment. The metric is the AUC on the test datasets labeled on the right vertical axis. To make the horizontal axis comparable across its domain, we linearize the AUC using the logit (i.e., the logarithm of the odds) in base 2, shown on the bottom axis. The original AUC values appear on the top axis.

The plots reveal two advantages of the self-supervised pipelines: first, performances (means and medians) tend to be higher; second, the variability (width of the boxes) tends to be smaller. That shows the self-supervised pre-training's ability not only to improve the results but also to make them more stable. No consistent advantage in terms of trend improvement (mean, median) is evident among the different self-supervised pipelines, but in terms of variability reduction, the double-pre-trained pipelines (SSL $\rightarrow$ SCL/UCL $\rightarrow$ FT) appear to have a slight advantage.

5.1.5 Low-Training Data Scenario

These results follow the same protocol as those in the previous section but with drastically reduced train datasets. The results appear in the middle and bottom-most plots in Figure 5.6, for 10% (1480 samples) and 1% (148 samples), respectively, of the original train dataset. Other than this restriction, the interpretation of the plots is the same as in the previous section. The results are much noisier than the full-data experiments: this is intrinsic to the smaller training sets, but the random choice of training subsets also contributes to increased variability.

Again, the self-supervised pipelines appear advantageous in trend improvement (mean, median) and variability reduction. However, here the advantage of the double-pre-trained pipelines (SSL $\rightarrow$ SCL/UCL $\rightarrow$ FT) seems more decisive, especially for the lowest data regimen, where

it improves both in trend and variability. As discussed in the conclusions, such variability reduction is critical for the soundness of the deployment of low-data models.

5.1.6 Implementation Details

We use PyTorch-Lightning[1] for the main development, PyContrast[2] for the self-supervised pre-trainings, and Comet.ML[3], and Weights & Biases[4] for experiment management. All experiments ran in a single RTX 5000 GPU, except for the SSL $\rightarrow$ UCL/SCL $\rightarrow$ FT pipelines on a 512-batch size, which required two Quadro RTX 8000 GPUs. The ResNet-50 supervised pre-trained weights on ImageNet used on the baseline came from torchvision. For the z-normalization, we use the ImageNet RGB channel means $(0.485, 0.456, 0.406)$, and standard deviations $(0.228, 0.224, 0.225)$.

The original self-supervised models were pre-trained by their authors as follows.

- BYOL: batch size = 4096, epochs = 1000 (temperature parameter unused at pre-trained);
- InfoMin: batch size = 256, temperature = 0.07, epochs = 800;
- MoCo: batch size = 256, temperature = 0.07, epochs = 800;
- SimCLR: batch size = 4096, temperature = 0.1, epochs = 800;
- SwAV: batch size = 4096, temperature = 0.1, epochs = 800.

All models are pre-trained on ImageNet.

The source code used in this work, in addition to detailed descriptions of the data and instructions to reproduce our experiments, is available on our source-code repository https://github.com/VirtualSpaceman/ssl-skin-lesions.

5.2 The General Case & Other Medical Applications

5.2.1 Evaluation Metrics & Datasets

We use balanced accuracy as the primary metric to evaluate the model's performance on several different tasks under many datasets and classes. We preferred such a metric because all datasets are unbalanced, and it facilitates the comparison of all experiments by standardizing the reported metric. Even if some datasets consist only of two classes and AUC presents as a better metric for binary classification, we kept the balanced accuracy to better group the experiments and made them comparable.

We list and describe the datasets we use in both medical and natural images application. We highlight our contribution of surveying and organizing several datasets available in the literature and setup in- and out-distribution scheme for all applications. To the best of our knowledge,

[1] https://github.com/PyTorchLightning/pytorch-lightning
[2] https://github.com/HobbitLong/PyContrast
[3] https://www.comet.ml
[4] https://wandb.ai

Figure 5.6: Results for the second round of experiments, with a systematic comparison of the pipelines labeled on the left vertical axis at the datasets labeled on the right vertical axis. The top, middle, and bottom plots show results for 100%, 10% and, 1% of the training data, respectively. Individual measurements of each boxplot appear as small black dots, whose means appear as larger red dots. In general, self-supervised pre-trained improves trends (medians, means) and reduces variability in both full-data and low-data scenarios.

this is the first work to organize and survey several datasets to set up in- and out-distribution, including general-purpose and medical image applications and self-supervision.

For the rest of the medical applications, we use the following datasets:

PatchCamelyon [114]: It contains histopathologic scans of lymph node sections extracted from the whole-slide images in the study at Veeling et al. [114]. All of the slides are annotated by expert pathologists. If the center of a patch contains at least one pixel of tumor tissue, it will be a malignant sample. The data version we use is the one for WILDS Benchmark [73]. The original train set consists of $220,025$ patches of size 96×96 with binary labels indicating whether there is a tumor or not. The authors made available the center information in which each image was taken. In this way, we can split the dataset by leaving one hospital as a test (out-of-distribution) or using patient ID to properly split without contaminating train, validation, and test splits (in-distribution).

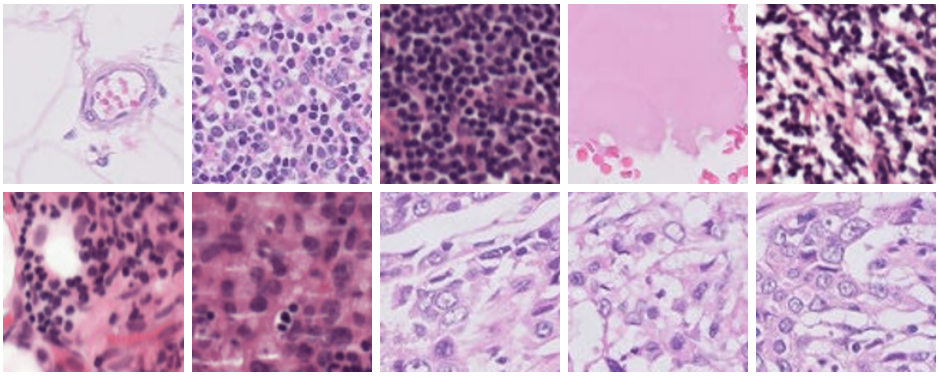

Figure 5.7: Samples from PatchCamelyon dataset. The first row shows only benign samples, while the bottom row only malignant samples.

BreakHis[5]: It is composed of $9,109$ microscopic images of breast tumor tissue collected from 82 patients using different magnifying factors. It is a binary dataset consisting of benign $(2,480)$ and malignant $(5,429)$ tumors. The database was collected in collaboration with the P&D Laboratory -– Pathological Anatomy and Cytopathology, Paraná, Brazil. We resized all images to 299×299.

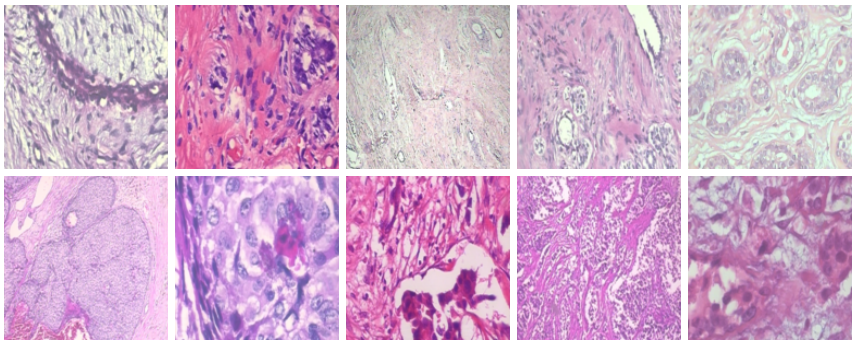

Figure 5.8: Samples from BreakHis dataset.The first row shows only benign samples, while the bottom row only malignant samples.

[5] https://www.kaggle.com/ambarish/breakhis

ICIAR2018_BACH[6]: The image dataset consists of 400 stained breast histology microscopy images. Each image is labeled with one of the four balanced classes: normal, benign, in situ carcinoma, and invasive carcinoma, where a class is defined as a predominant cancer type in the image. Two medical experts performed the image annotation. We use the data for testing only for models trained on the BreakHis dataset. We removed the "normal" samples and relabeled the carcinomas to a single malignant class to match the binary classification in BreakHis. We resized all images to 299×299.

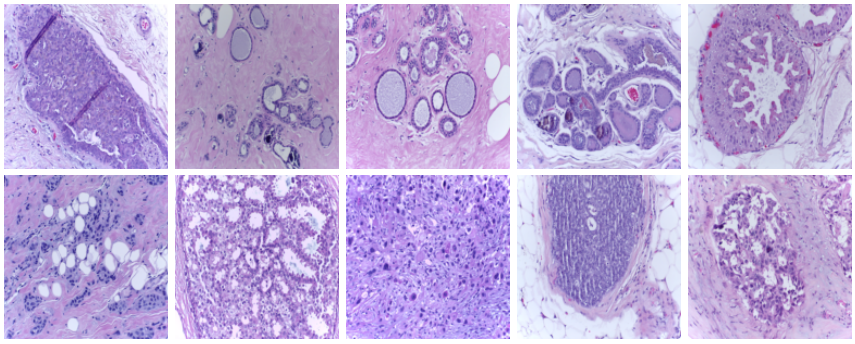

Figure 5.9: Samples from ICIAR2018 dataset. The first row shows only benign samples, while the bottom row only malignant samples.

BrainTumor-Cheng [29]: It consists of a brain magnetic resonance imaging (MRI) dataset acquired from Nanfang Hospital, Guangzhou, China, and General Hospital, Tianjin Medical University, China, from 2005 to 2010. The authors collected $3,064$ slices from 233 patients, containing 708 meningiomas, $1,426$ gliomas, and 930 pituitary tumors. We use it as the main dataset to train in a brain tumor classification scenario. We resized all images to 299×299.

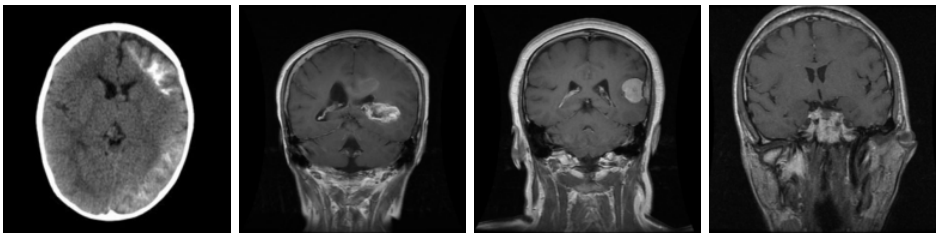

Figure 5.10: Samples from BrainTumor-Cheng dataset.

NINS [15]: It consists of $5,285$ MRI images from brain scans. The data was collected in collaboration with the National Institute of Neuroscience & Hospitals (NINS) of Bangladesh [15]. In total, they have 37 classes, but we only keep those present in our training set. It contains 76 gliomas, and 76 meningiomas, 76 pituitary tumors. We use this dataset as an out-of-distribution test for models trained on BrainTumor-Cheng. We resized all images to 299×299.

For general-purpose case, we use the following datasets:

[6] https://iciar2018-challenge.grand-challenge.org

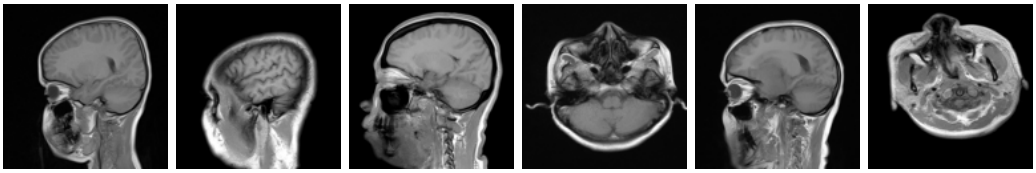

Figure 5.11: Samples from NINS dataset.

NICO [54]: It is a dataset of natural images essentially designed for out-of-distribution image classification. The basic idea is to label images with both classes and contexts. For example, in the category of "monkey" images are divided into different contexts such as "woods", "snow", "trees", meaning the "monkey" is in the woods, in the snow, or on trees. We can easily design an out-of-distribution setting with these contexts by training a model in some contexts and testing it in other unseen contexts. There are two superclasses available: Animal and Vehicle, with 10 classes for Animal and 9 classes for Vehicle. Each class has 9 or 10 contexts. The average number of images per class is about 1300 images. In total, NICO contains 19 classes, 188 contexts and nearly $25,000$ images.

Figure 5.12: Samples from NICO dataset.

CIFAR [76]: It is a collection of natural images largely used in general computer vision to benchmark models. Instead of working with its most famous versioning (CIFAR-10 or CIFAR-100), we use the superclass information to create in- and out-of-distribution schemes. For example, images originally labeled as "bee" or "butterfly" are relabeled as the superclass corresponding to "insects"; the full list showing the class-superclass correspondence is at CIFAR website[7]. The dataset contains $60,000$ samples and 20 superclasses ("aquatic mammals", "fish", "flowers", "food containers", "fruit and vegetables", "household electrical devices", "household furniture", "insects", "large carnivores", "large man-made outdoor things", "large natural outdoor scenes", "large omnivores and herbivores", medium-sized mammals", "non-insect invertebrates", "people", "reptiles", "small mammals", "trees", "vehicles 1", "vehicles 2").

Now, we describe how we create the in- and out-distribution splits for each set of applications.

For NICO, we leveraged the available context information for each class to build such in- and out-distribution scenarios. We randomly sampled 20% of the original dataset as a fixed

[7] https://www.cs.toronto.edu/~kriz/cifar.html

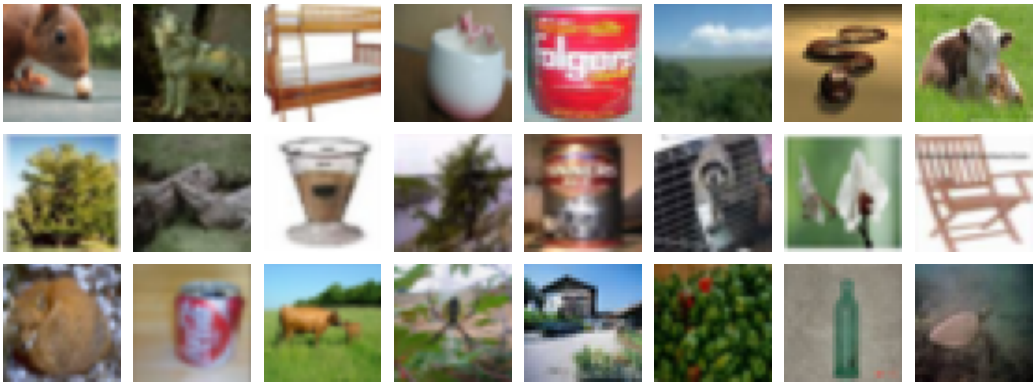

Figure 5.13: Samples from CIFAR dataset.

test set in the in-distribution scheme. We certified that all context for each class appears in the sampled test set. We use the remaining data to create stratified training and validation splits for both full and low-training data schemes. In the out-distribution case, we list all contexts for each class, and then we remove one random context out and reserve the removed samples as the test set. We use the remaining data to create splits for training and validation. We repeat this step-by-step by both Vehicle and Animal superclasses.

In CIFAR, we did a similar procedure as mentioned in NICO, but we leveraged the superclass information. We take 20% of the full data as the test set in the in-distribution scheme. We certified that we included samples for each class inside each superclass. In the out-distribution set, we list all superclasses and all the five classes belonging to each one and then take 1 out of 5 classes for each superclass and reserve those samples as a test set. We use the remaining data to generate stratified splits for training and validation.

For the experiments regarding the medical applications, we also randomly sampled 20% of the entire dataset and kept it as the in-distribution set since we drew samples from the same data distribution as training. We use the remaining data to split training and validation for full- and low-data experiments. For all applications, we use an external dataset freely available in the literature to assess the out-of-distribution performance. We purposefully take the datasets in which the labels between the training and testing are the same, but they differ in many aspects, such as capture device, medical protocol, population, age, or even country in the images was taken. Such difference in data distribution is expected in real-world applications, and the models will inevitably encounter such situations.

Table 5.4 shows the datasets, classification task, imagery, number of classes and size, and which dataset is used to assess out-of-distribution performance.

Dataset	Classification task	Imagery	Size	#Classes	Out-of-Distribution test
PatchCamelyon17	Lymph Node	Histopathologic	$335,996$	2	PatchCamelyon17
BreakHis	Breast cancer	Histopathologic	$7,909$	2	ICIAR2018_BACH
BrainTumor – Cheng	Brain tumor	Magnetic Resonance	$3,264$	4	NINS
CIFAR20	Natural image	Scrapped from the web	$60,000$	20	CIFAR20
NICO – Animal	Natural image	Scrapped from the web	$13,073$	10	NICO – Animal
NICO – Vehicle	Natural image	Scrapped from the web	$11,698$	9	NICO – Vehicle

Table 5.4: Description of the datasets used in both general-purpose and medical applications.

5.2.2 Pipeline's Hyperparameters

SUP → FT & SSL → FT

Initially, we decided to use the same best parameters for fine-tuning we found for skin lesion analysis. We thought such a parameter set would also be a good fit in other applications. Instead, we faced huge performance variation across datasets and pre-trained models, i.e., the same model's parametrization might not be adequate for two (or more) distinct datasets. Such variance in performance made us again look at the literature for large-scale studies in transfer learning. Inspired by Kornblith et al. [75], we decided to perform a grid search on learning rate and weight decay parameters since they observed a large performance correlation between those two [75]. Thus, our grid consists of 7 logarithmically spaced learning rates between 10^{-5} and 10^{-1} and 7 logarithmically spaced weight decay to learning rate ratios between 10^{-6} and 10^{-3} [75]. We always evaluate in the proper validation set to avoid using privileged information and inflating our models' performances.

We fine-tuned all models for 100 epochs at a batch size of varying the batch size in either 32 if the number of samples is below 15k or 128 otherwise. The learning rate and weighted decay were sampled from the grid. We use the SGD optimizer with a momentum of 0.9 and a cosine decay scheduler. We monitor the validation loss during training and take the model that presented the minimum value. We kept the early stopping with the patient of 22 epochs — the same value as in skin lesion experiments.

5.2.3 Low-Data and Out-of-Distribution Performance

As mentioned in Section 5.2.1, we use balanced accuracy as a standard metric to report all results. We optimized the hyperparameters for each method, dataset, and training percentage to create a fair scenario for all methods. The results are in Figures 5.14, 5.15, 5.16, 5.17, 5.18, and 5.19. Again, we use boxplots to show the model's performance variations, but we arranged the plots in the following way: the x-axis pictures, the pre-training method in fine-tuning, and the y-axis depict the balanced accuracy. We organized each plot in two rows: in-distribution (top) and out-of-distribution (bottom); and three columns showing the training percentages: leftmost (1%), middle (10%), and rightmost (100%). Such plots allow us to answer the Research Question 2: "*How do self-supervised models pre-trained on ImageNet perform in medical imaging compared to supervised pre-trained ImageNet models?*", and Research Question 3: "*How do self-supervised models perform when only a few samples are available for training and when out-of-distribution test datasets for medical and general-purpose applications?*".

We make two main observations by analyzing the individual results for each dataset. First, the performance on NICO subsets differs a lot from the remaining plots. Such behavior leads us to assume that bimodal behavior is associated with the results. Second, it is hard to determine if self-supervised pre-training is advantageous or in which contexts they are superior. Our subsequent analysis intends to remove the difficulty regarding the dataset and focus on giving an overall recommendation, i.e., if we would recommend a pre-training method which one would we take?

Figures 5.20 and 5.21 show the general performances, after removing the dataset difficulty. We group the plot by method and training percentage. We preferred to isolate the NICO's in-

Figure 5.14: Box plots showing the model's performance for the NICO – Animal set. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (1%, 10%, and 100%) in columns and in-/out-distribution performances (rows). We observe that the supervised baseline is superior in low-data scenarios for in- and out-of-distribution. In the full-data case, all methods excel at the target task with no significant difference in performance.

Figure 5.15: Box plots showing the model's performance for the NICO – Vehicle set. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (1%, 10%, and 100%) in columns and in-/out-distribution performances (rows). Again, we observe that the supervised baseline is superior in low-data scenarios for in- and out-of-distribution. In the full-data case, all methods excel at the target task with no significant difference in performance.

fluence from the other sets due to our hypothesis about the bimodal data distribution. Then, we normalize the data considering each dataset, training percentage, and split. It permits an investigation of which models are above or below the average performance. Then, we aggregate all differences for each training percentage and sum them up. This procedure aims to understand

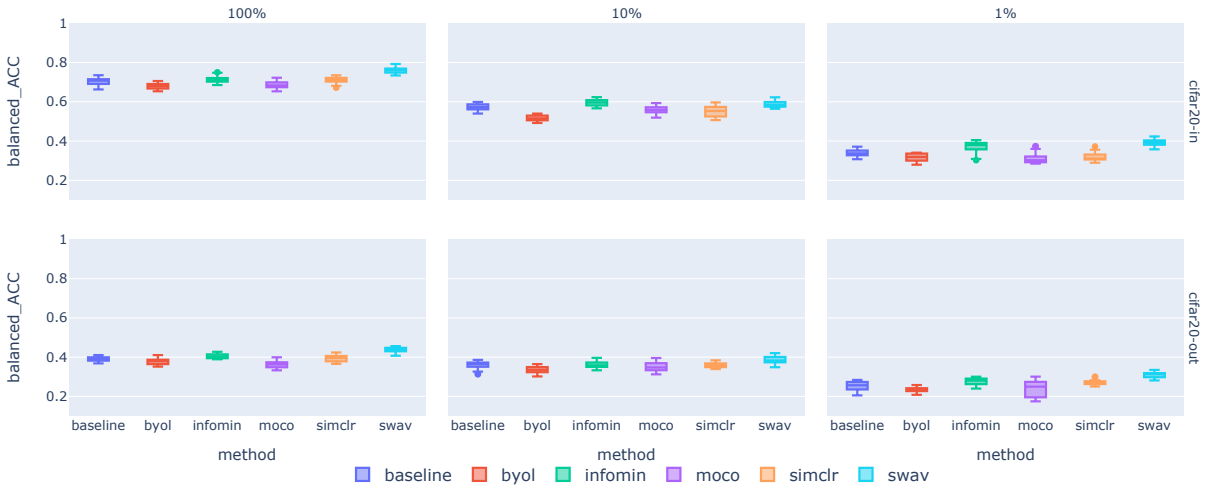

Figure 5.16: Box plots showing the model's performance for the CIFAR-20 set. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (1%, 10%, and 100%) in columns and in-/out-distribution performances (rows). We observe that some self-supervised are slightly superior to the supervised baseline, especially the SwAV method in both in- and out-distribution.

Figure 5.17: Box plots showing the model's performance for the BreakHis set. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (100%, 10%, and 1%) in columns and in-/out-distribution performances (rows). Again, we observe that some self-supervised are slightly superior to the supervised baseline in both in- and out-distribution. All methods experienced large variance on 1% scenario for in-distribution but low variance on out-of-distribution performance.

which pre-training scheme gives better performance, independent of the dataset. Suppose a method has a positive-sum means that the method was above average in most scenarios compared to the other competitors. In this case, separating the influence of the NICO dataset was essential to interpret the results better since it presents a unique behavior.

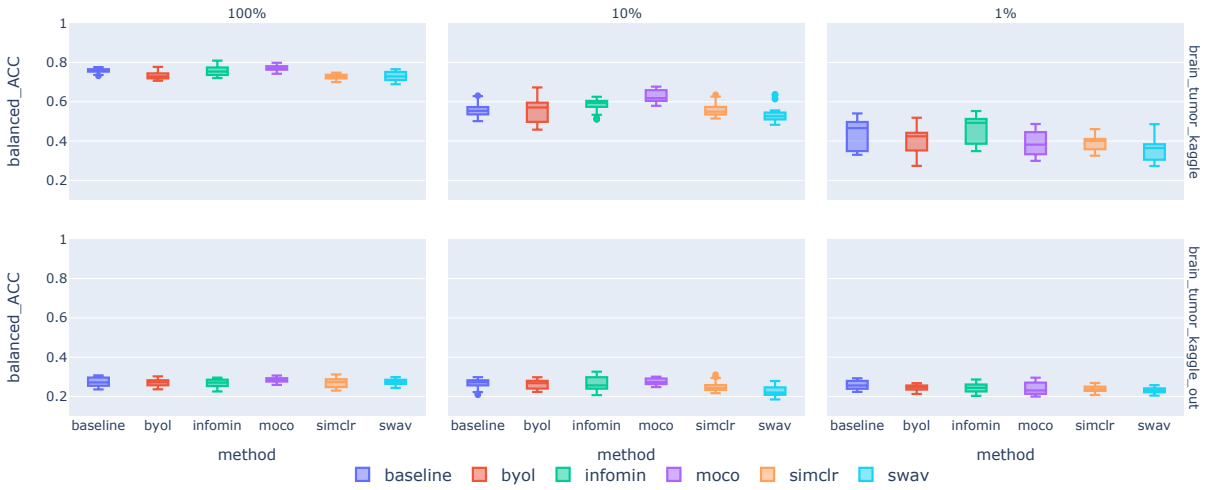

Figure 5.18: Box plots showing the model's performance for the BrainTumor set. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (100%, 10%, and 1%) in columns and in-/out-distribution performances (rows). Again, some self-supervised present slightly superior in-distribution performance than the supervised baseline. Surprisingly, none of the training percentages helped to improve the out-of-distribution performance, which is kept essentially the same.

Figure 5.19: Box plots showing the model's performance for the PatchCamyleon17 set. We found no clear winner in in-distribution performance. All results are in some sort equally good, and even in low-data, the performance is accurate, indicating the problem is very easy for all methods. However, as the training set grows, we observe a drop in the out-of-distribution performance. Such behavior might indicate that all models are overfitted, and using the full data can detriment the out-of-distribution performance.

Unfortunately, our work lacks comparison with the current state of the art due to the experimental design conducted. As we sacrifice part of our dataset to set up the in- and out-distribution case, we can not guarantee using the official train, validation, and test splits for each dataset.

We made all data available in this Master Dissertation to facilitate further reproduction of our

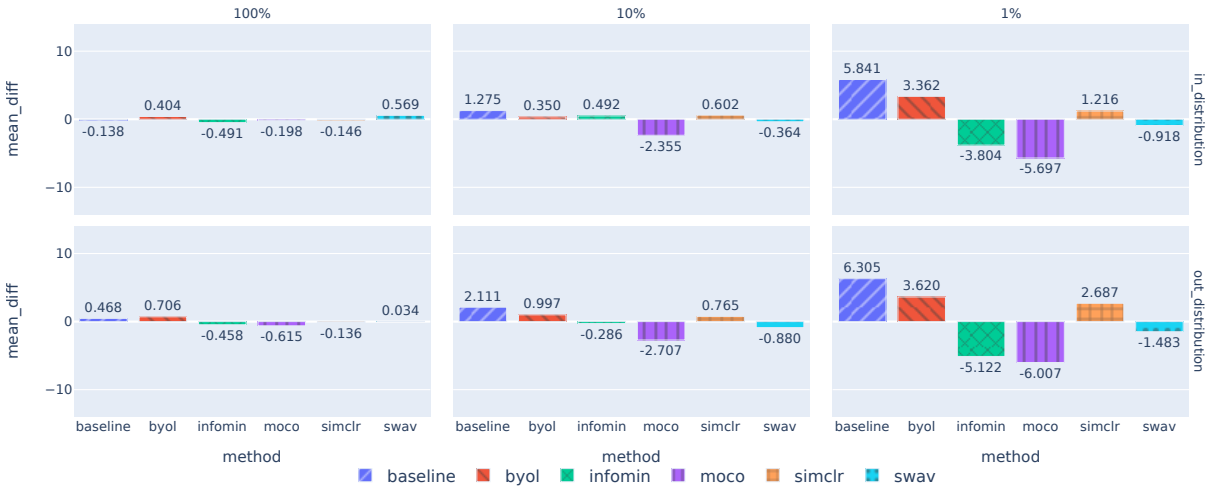

Figure 5.20: The sum of the differences according to the mean performance for the NICO dataset. We isolate the NICO influence due to the bimodal data distribution hypothesis. We labeled the x-axis according to the pre-training method, and the y-axis reports the balanced accuracy. We organized the plots according to the training percentages (100%, 10%, and 1%) in columns and in-/out-distribution performances (rows). We observe that the supervised approach has a much higher cumulative summation than any other self-supervised method for scenarios with 1% of the data. This advantage becomes smaller in the 10% data scenario and practically disappears when we observe the full-data scenario.

Figure 5.21: The sum of the differences according to the mean performance for CIFAR-20 and all medical datasets. We observe that some self-supervised methods have a superior accumulated sum over the baseline, but no pre-training scheme showed a consistent advantage over the baseline. The major difference appears in out-of-distribution, where the differences are slightly favorable to self-supervision models, but no major difference comparing the in-distribution performance.

Chapter 6

Conclusion

In this chapter, we review our findings, covering the major topics discussed in this Master Dissertation. We discuss future directions to improve our results and critically analyze our experimental design.

As the topic of self-supervised has become famous and emerged as an alternative way to train neural network models, the literature points in a direction where self-supervised methods tend to perform comparable or even better than supervised methods on some tasks. This Master Dissertation investigated the impact of using five self-supervised methods on several classification tasks, encompassing medical and general contexts. We included scenarios when limited data is available for training and evaluated several out-of-distribution datasets. Our work is the first that has performed this organization and evaluation on out-of-distribution datasets focusing on medical context for classification problems and involving self-supervised methods.

We split our experimental protocol into two fronts. We first fine-tuned five self-supervised methods against a supervised baseline on the skin lesion task in scenarios when only a subset of the original data is available and in out-of-distribution cases. We also investigated using a new pre-training step before performing the fine-tune and observed that it showed better results in out-of-distribution scenarios and little impact on the in-distribution performance. Our subsequent investigation explores the experimental protocol in other medical and natural scenarios. The results suggest a slight advantage in using self-supervised methods for low-data and out-of-distribution scenarios. No self-supervised method proved consistently better. However, we did not find any significant difference between supervised and self-supervised methods for the evaluated datasets is a positive point: we can use either pre-trained model's method as initialization for fine-tuning, but the self-supervised methods one does not require millions of labeled examples for pre-training.

After conducting the experiments, we were able to answer the research questions proposed in Chapter 1:

Q1. Is there any benefit in using self-supervised models instead of supervised models as a starting point for fine-tuning?
In some cases, yes, but none of the investigated self-supervised consistently boosted performance. Several papers report a considerable gain from using pre-trained self-supervised models, but these gains come at excessive optimization on privileged sets, unfair comparisons, or even too few runs to check for variability. When we systemat-

ically evaluate the methods over several scenarios, the gains shown are unique to the pre-training technique. Our results are on par with recent benchmarks in literature for natural images [43,47,88,117] and medical applications [60,110] for classification tasks, with ours standing out in covering both applications and performance evaluation of in- and out-of-distribution scenarios.

Q2. How do self-supervised models pre-trained on ImageNet perform in medical imaging compared to supervised pre-trained ImageNet models?
Supervised and self-supervised models showed similar performance on medical classification tasks when evaluating all methods using the ResNet-50 ($1\times$) as a backbone and our hyperparameter choices. All methods showed a large variance in a low-data scenario when less than $5,000$ samples were available. Some self-supervised models showed a superior performance in out-of-distribution scenarios (Figures 5.17, 5.18, and 5.19), depending on the training data regime (1%, 10%, or 100%).

Q3. How do self-supervised models perform when only a few samples are available for training and when out-of-distribution test datasets for medical and general-purpose applications?
As in the previous question, supervised and self-supervised models showed similar performance when posing pre-training schemes on medical and general-purpose classification tasks. However, all investigated methods had a huge performance drop when the out-of-distribution test set. Such behavior indicates that all pre-trained models suffer when distribution shifts exist in the test set and lack high generalization capabilities. In some applications, such as in BreakHis and BrainTumor, we observed no improvement in the out-of-distribution performance regardless of the full- or low-data regime.

6.1 Limitations and Future Work

We believe our work contributed to the medical and general-purpose computer vision field. However, our discussion and experimental design have some drawbacks. Our protocol addressed only the ResNet-50 backbone, and our conclusions consider it the primary encoder. It is known that commonly deeper models yield the best results. The same behavior may not hold, or new trends may appear if different backbones are used. Therefore, future work would also investigate the impact of other backbones and if the results hold.

Another limitation of our work is applying the experimental protocol in general-purpose and medical imaging. To reduce the exhaustive number of experiments, we decided to optimize the hyperparameters taking into account only one split and replicating the best parameters for the remaining splits. We did the same procedure for all datasets and percentages. Ideally, we should do this search for each split and not just replicate the best set of parameters. Due to the random process of splitting the data into splits, neural networks may deteriorate depending on the dataset used. Therefore, for some splits, it may not have been the best combination, and as a consequence, poor results were reported for that split.

Due to the high computational cost, we did not evaluate applying an extra pre-training step in-domain. We found benefits in skin lesions in performing this extra step, but due to the time

required, we chose not to bring this procedure to a more extensive evaluation. We hypothesize that we would observe similar gains in the target tasks since the procedure can be used with labeled or unlabeled data. In addition, we performed such evaluation only on image classification tasks. However, the results may show entirely different behavior when further explored in other tasks, such as the segmentation task. One research direction is to extend our work to other tasks, similar to Hosseinzadeh et al. [60], but including out-of-distribution evaluation; or include Transformers-based self-supervised models [69].

There are several future directions to employ self-supervised in medical applications. We reviewed all pretext tasks carefully crafted by computer vision experts and involved many trials and experiments to let the pretext task peak performance. A future direction is to formulate the "task finding" as an optimization problem similar to what happens in neural architecture search. If given enough examples, it might be possible to find the optimal pretext task. This process would consist of creating new pipelines of pretext tasks and comparing these pretext pipelines with related target tasks. The researcher would comprehend which pretext tasks are the best ones to use as a starting point for a given task and learn what would work on the target application.

Our results showed that even with each model at its peak, none of the five investigated self-supervised methods have proven consistently better than the supervised baseline in all target applications, but the best one varied depending on the target task. However, we decided to look at this behavior positively: if there is no difference between supervised and self-supervised methods in performance, then self-supervised training may be preferable because it eliminates the need for labeled data in the pre-training step. Moreover, self-supervised learning allows the original pre-training using labeled and unlabeled data from the target task before the fine-tuning step. We hypothesize that such behavior occurs due to the original self-supervised pre-training scheme adding difficulty in capturing the low inter-class and intra-class variation details in medical applications. However, an intriguing way to continue this work is to understand why these models perform as supervised models and the impact of different pre-training forms on various tasks. Some directions are to investigate how pre-training impacts the investigating model's invariances [38] or transfer capabilities [6, 51, 66] under a theoretical perspective.

Bibliography

[1] Christian Abbet, Inti Zlobec, Behzad Bozorgtabar, and Jean-Philippe Thiran. Divide-and-rule: self-supervised learning for survival analysis in colorectal cancer. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 480–489. Springer, 2020.

[2] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020.

[3] Mehdi Azabou, Mohammad Gheshlaghi Azar, Ran Liu, Chi-Heng Lin, Erik C Johnson, Kiran Bhaskaran-Nair, Max Dabagia, Bernardo Avila-Pires, Lindsey Kitchell, Keith B Hengen, et al. Mine your own view: Self-supervised learning through across-sample prediction. *arXiv preprint arXiv:2102.10106*, 2021.

[4] Shekoofeh Azizi, Basil Mustafa, Fiona Ryan, Zachary Beaver, Jan Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, Ting Chen, et al. Big self-supervised models advance medical image classification. *International Conference on Computer Vision*, 2021.

[5] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

[6] Yamini Bansal, Gal Kaplun, and Boaz Barak. For self-supervised learning, rationality implies generalization, provably. In *International Conference on Learning Representations*, 2020.

[7] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *International Conference on Learning Representations*, 2022.

[8] Alceu Bissoto, Michel Fornaciali, Eduardo Valle, and Sandra Avila. Generating high quality synthetic skin lesions for boosting automated screening. In *International Educational Symposium of the Melanoma World Society*, pages 43–43, 2018.

[9] Alceu Bissoto, Michel Fornaciali, Eduardo Valle, and Sandra Avila. (De)Constructing bias on skin lesion datasets. In *Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[10] Alceu Bissoto, Fábio Perez, Vinícius Ribeiro, Michel Fornaciali, Sandra Avila, and Eduardo Valle. Deep-learning ensembles for skin-lesion segmentation, analysis, classification: Recod titans at isic challenge 2018. *arXiv preprint arXiv:1808.08480*, 2018.

[11] Alceu Bissoto, Fábio Perez, Eduardo Valle, and Sandra Avila. Skin lesion synthesis with generative adversarial networks. In *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*, pages 294–302, 2018.

[12] Alceu Bissoto, Eduardo Valle, and Sandra Avila. Debiasing skin lesion datasets and models? not so fast. In *Conference on Computer Vision and Pattern Recognition Workshops*, pages 3192–3201, 2020.

[13] Alceu Bissoto, Eduardo Valle, and Sandra Avila. Gan-based data augmentation and anonymization for skin-lesion analysis: A critical review. In *Conference on Computer Vision and Pattern Recognition Workshops*, pages 1847–1856, 2021.

[14] Joseph Boyd, Mykola Liashuha, Eric Deutsch, Nikos Paragios, Stergios Christodoulidis, and Maria Vakalopoulou. Self-supervised representation learning using visual field expansion on digital pathology. In *International Conference on Computer Vision*, pages 639–647, 2021.

[15] Yusuf Brima, Mossadek Hossain Kamal Tushar, Upama Kabir, and Tariqul Islam. Deep transfer learning for brain magnetic resonance image multi-class classification. *arXiv preprint arXiv:2106.07333*, 2021.

[16] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision*, pages 132–149, 2018.

[17] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Conference on Neural Information Processing Systems*, 2020.

[18] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9650–9660, 2021.

[19] Levy Chaves, Alceu Bissoto, Eduardo Valle, and Sandra Avila. An evaluation of self-supervised pre-training for skin-lesion analysis. *arXiv preprint arXiv:2106.09229*, 2021.

[20] Kejiang Chen, Yuefeng Chen, Hang Zhou, Xiaofeng Mao, Yuhong Li, Yuan He, Hui Xue, Weiming Zhang, and Nenghai Yu. Self-supervised adversarial training. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2218–2222, 2020.

[21] Liang Chen, Paul Bentley, Kensaku Mori, Kazunari Misawa, Michitaka Fujiwara, and Daniel Rueckert. Self-supervised learning for medical image analysis using image context restoration. *Medical Image Analysis*, 58:101539, 2019.

[22] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, and Heewoo Jun. Generative pretraining from pixels. In *International Conference on Machine Learning.*, 2020.

[23] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020.

[24] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised gans via auxiliary rotation loss. In *Conference on Computer Vision and Pattern Recognition*, pages 12154–12163, 2019.

[25] Xiaocong Chen, Lina Yao, Tao Zhou, Jinming Dong, and Yu Zhang. Momentum contrastive learning for few-shot covid-19 diagnosis from chest ct images. *Pattern recognition*, 113:107826, 2021.

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

[27] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[28] Xuxin Chen, Ximin Wang, Ke Zhang, Roy Zhang, Kar-Ming Fung, Theresa C Thai, Kathleen Moore, Robert S Mannel, Hong Liu, Bin Zheng, et al. Recent advances and clinical applications of deep learning in medical image analysis. *arXiv preprint arXiv:2105.13381*, 2021.

[29] Jun Cheng, Wei Huang, Shuangliang Cao, Ru Yang, Wei Yang, Zhaoqiang Yun, Zhijian Wang, and Qianjin Feng. Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PloS one*, 10(10):e0140381, 2015.

[30] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Conference on Computer Vision and Pattern Recognition*, pages 539–546, 2005.

[31] Ozan Ciga, Tony Xu, and Anne Louise Martel. Self supervised contrastive learning for digital histopathology. *Machine Learning with Applications*, page 100198, 2021.

[32] Noel Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical ienaging (isbi), hosted by the international skin imaging collaboration (isic). In *International Symposium on Biomedical Imaging*, pages 168–172, 2018.

[33] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, pages 2292–2300, 2013.

[34] Jia Deng, Wei. Dong, Richard Socher, Li-Jai Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Conference on Computer Vision and Pattern Recognition*, 2009.

[35] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *International Conference on Computer Vision*, pages 1422–1430, 2015.

[36] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9):1734–1747, 2016.

[37] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *International Conference on Computer Vision*, 2021.

[38] Linus Ericsson, Henry Gouk, and Timothy M Hospedales. Why do self-supervised models transfer? investigating the impact of invariance on downstream tasks. *arXiv preprint arXiv:2111.11398*, 2021.

[39] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[40] Michel Fornaciali, Sandra Avila, Micael Carvalho, and Eduardo Valle. Statistical learning approach for robust melanoma screening. In *SIBGRAPI Conference on Graphics, Patterns and Images*, pages 319–326, 2014.

[41] Michel Fornaciali, Micael Carvalho, Flávia Vasques Bittencourt, Sandra Avila, and Eduardo Valle. Towards automated melanoma screening: Proper computer vision & reliable results. *arXiv preprint arXiv:1604.04024*, 2016.

[42] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[43] Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Tizian Thieringer, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Partial success in closing the gap between human and machine vision. *Advances in Neural Information Processing Systems*, 2021.

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

[45] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[46] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.

[47] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *International Conference on Computer Vision*, pages 6391–6400, 2019.

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

[49] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

[50] Fatemeh Haghighi, Mohammad Reza Hosseinzadeh Taher, Zongwei Zhou, Michael B Gotway, and Jianming Liang. Learning semantics-enriched representation via self-discovery, self-classification, and self-restoration. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 137–147. Springer, 2020.

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

[52] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[53] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[54] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, page 107383, 2020.

[55] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192, 2020.

[56] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations*, 2019.

[57] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[58] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *International Conference on Learning Representations*, 2019.

[59] Olle G Holmberg, Niklas D Köhler, Thiago Martins, Jakob Siedlecki, Tina Herold, Leonie Keidel, Ben Asani, Johannes Schiefelbein, Siegfried Priglinger, Karsten U Kortuem, et al. Self-supervised retinal thickness prediction enables deep learning from unlabelled data to boost classification of diabetic retinopathy. *Nature Machine Intelligence*, 2(11):719–726, 2020.

[60] Mohammad Reza Hosseinzadeh Taher, Fatemeh Haghighi, Ruibin Feng, Michael B Gotway, and Jianming Liang. A systematic benchmarking analysis of transfer learning for medical image analysis. In *Domain Adaptation and Representation Transfer, and Affordable Healthcare and AI for Resource Diverse Global Health*, pages 3–13. Springer, 2021.

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

[62] Rui Huang, Wenju Xu, Teng-Yok Lee, Anoop Cherian, Ye Wang, and Tim Marks. Fx-gan: Self-supervised gan learning via feature exchange. In *IEEE Winter Conference on Applications of Computer Vision*, pages 3194–3202, 2020.

[63] Ali Jahanian, Xavier Puig, Yonglong Tian, and Phillip Isola. Generative models as a data source for multiview representation learning. *arXiv preprint arXiv:2106.05258*, 2021.

[64] Amir Jamaludin, Timor Kadir, and Andrew Zisserman. Self-supervised learning for spinal mris. In *Medical Image Computing and Computer Assisted Interventions Workshops*, pages 294–302. 2017.

[65] Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Conference on Computer Vision and Pattern Recognition*, pages 2733–2742, 2018.

[66] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *International Conference on Learning Representations*, 2022.

[67] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[68] Jeremy Kawahara, Sara Daneshvar, Giuseppe Argenziano, and Ghassan Hamarneh. Seven-point checklist and skin lesion classification using multitask multimodal neural nets. *IEEE Journal of Biomedical and Health Informatics*, 23(2):538–546, 2019.

[69] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM Computing Surveys*, 2021.

[70] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[71] Dahun Kim, Donghyeon Cho, Donggeun Yoo, and In So Kweon. Learning image representations by completing damaged jigsaw puzzles. In *Winter Conference on Applications of Computer Vision*, pages 793–802. IEEE, 2018.

[72] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[73] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664, 2021.

[74] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[75] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Conference on Computer Vision and Pattern Recognition*, pages 2661–2671, 2019.

[76] Alex Krizhevsky and George Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

[77] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *European Conference on Computer Vision*, pages 577–593. Springer, 2016.

[78] Jiajun Li, Tiancheng Lin, and Yi Xu. Sslp: Spatial guided self-supervised learning on pathological images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 3–12. Springer, 2021.

[79] Xiaomeng Li, Mengyu Jia, Md Tauhidul Islam, Lequan Yu, and Lei Xing. Self-supervised feature learning via exploiting multi-modal data for retinal disease diagnosis. *IEEE Transactions on Medical Imaging*, 39(12):4023–4033, 2020.

[80] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. A survey of transformers. *arXiv preprint arXiv:2106.04554*, 2021.

[81] Stuart Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[82] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[83] Afonso Menegola, Julia Tavares, Michel Fornaciali, Lin Tzy Li, Sandra Avila, and Eduardo Valle. Recod titans at isic challenge 2017. *arXiv preprint arXiv:1703.04819*, 2017.

[84] John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, pages 7721–7735, 2021.

[85] Jovana Mitrovic, Brian McWilliams, Jacob Walker, Lars Buesing, and Charles Blundell. Representation learning via invariant causal mechanisms. *International Conference on Learning Representations*, 2021.

[86] Daniel I. Morís, Álvaro S. Hervella, José Rouco, Jorge Novo, and Marcos Ortega. Context encoder self-supervised approaches for eye fundus analysis. In *International Joint Conference on Neural Networks*, pages 1–8, 2021.

[87] Hassan Muhammad, Carlie S. Sigel, Gabriele Campanella, Thomas Boerner, Linda M. Pak, Stefan Büttner, Jan N. M. IJzermans, Bas Groot Koerkamp, Michael Doukas, William R. Jarnagin, Amber L. Simpson, and Thomas J. Fuchs. Unsupervised subtyping of cholangiocarcinoma using a deep clustering convolutional autoencoder. In *Medical Image Computing and Computer Assisted Intervention*, pages 604–612, 2019.

[88] Alejandro Newell and Jia Deng. How useful is self-supervised pretraining for visual tasks? In *Conference on Computer Vision and Pattern Recognition*, pages 7345–7354, 2020.

[89] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *European Conference on Computer Vision*, pages 69–84, 2016.

[90] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Conference on Computer Vision and Pattern Recognition*, pages 9359–9367, 2018.

[91] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[92] Andre GC Pacheco, Gustavo R Lima, Amanda S Salomão, Breno Krohling, Igor P Biral, Gabriel G de Angelo, Fábio CR Alves Jr, José GM Esgario, Alana C Simora, Pedro BC Castro, et al. Pad-ufes-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones. *Data in brief*, 32:106221, 2020.

[93] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Conference on Computer Vision and Pattern recognition*, pages 2536–2544, 2016.

[94] Fábio Perez, Sandra Avila, and Eduardo Valle. Solo or ensemble? choosing a cnn architecture for melanoma classification. In *Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[95] Fábio Perez, Cristina Vasconcelos, Sandra Avila, and Eduardo Valle. Data augmentation for skin lesion analysis. In *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*, pages 303–311. 2018.

[96] Adalberto Claudio Quiros, Roderick Murray-Smith, and Ke Yuan. Pathologygan: Learning deep representations of cancer tissue. *Journal of Machine Learning for Biomedical Imaging*, pages 1–48, 2021.

[97] Vinicius Ribeiro, Sandra Avila, and Eduardo Valle. Less is more: Sample selection and label conditioning improve skin lesion segmentation. In *Conference on Computer Vision and Pattern Recognition Workshops*, pages 3182–3191, 2020.

[98] Veronica Rotemberg, Nicholas Kurtansky, Brigid Betz-Stablein, Liam Caffery, Emmanouil Chousakos, Noel Codella, Marc Combalia, Stephen Dusza, Pascale Guitera, David Gutman, et al. A patient-centric dataset of images and metadata for identifying melanomas using clinical context. *Scientific data*, 8(1):1–8, 2021.

[99] Saeed Shurrab and Rehab Duwairi. Self-supervised learning methods and applications in medical imaging analysis: A survey. *arXiv preprint arXiv:2109.08685*, 2021.

[100] Hari Sowrirajan, Jingbo Yang, Andrew Y Ng, and Pranav Rajpurkar. Moco pretraining improves representation and transferability of chest x-ray models. In *Medical Imaging with Deep Learning*, pages 728–744, 2021.

[101] Anuroop Sriram, Matthew Muckley, Koustuv Sinha, Farah Shamout, Joelle Pineau, Krzysztof J Geras, Lea Azour, Yindalon Aphinyanaphongs, Nafissa Yakubova, and William Moore. Covid-19 deterioration prediction via self-supervised representation learning and multi-image prediction. *arXiv preprint arXiv:2101.04909*, 2021.

[102] Marco Tagliasacchi, Beat Gfeller, Félix de Chaumont Quitry, and Dominik Roblek. Pretraining audio representations with self-supervision. *IEEE Signal Processing Letters*, 27:600–604, 2020.

[103] Nima Tajbakhsh, Yufei Hu, Junli Cao, Xingjian Yan, Yi Xiao, Yong Lu, Jianming Liang, Demetri Terzopoulos, and Xiaowei Ding. Surrogate supervision for medical image analysis: Effective deep learning from limited quantities of labeled data. In *International Symposium on Biomedical Imaging*, pages 1251–1255. IEEE, 2019.

[104] Aiham Taleb, Winfried Loetzsch, Noel Danz, Julius Severin, Thomas Gaertner, Benjamin Bergner, and Christoph Lippert. 3d self-supervised methods for medical imaging. *Advances in Neural Information Processing Systems*, 33:18158–18172, 2020.

[105] Alex Tamkin, Mike Wu, and Noah Goodman. Viewmaker networks: Learning views for unsupervised representation learning. In *International Conference on Learning Representations*, 2020.

[106] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European Conference on Computer Vision*, 2020.

[107] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. In *Conference on Neural Information Processing Systems*, 2020.

[108] Yuandong Tian, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. In *International Conference on Machine Learning*, pages 10268–10278, 2021.

[109] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Conference on Computer Vision and Pattern Recognition*, pages 1521–1528, 2011.

[110] Tuan Truong, Sadegh Mohammadi, and Matthias Lenga. How transferable are self-supervised features in medical image classification tasks? In *Machine Learning for Health*, pages 54–74, 2021.

[111] Eduardo Valle, Michel Fornaciali, Afonso Menegola, Julia Tavares, Flávia Vasques Bittencourt, Lin Tzy Li, and Sandra Avila. Data, depth, and design: Learning reliable models for skin lesion analysis. *Neurocomputing*, 383:303–313, 2020.

[112] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European Conference on Computer Vision*, pages 268–285, Cham, 2020.

[113] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[114] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 210–218. Springer, 2018.

[115] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, pages 1096–1103, 2008.

[116] Yen Nhi Truong Vu, Richard Wang, Niranjan Balachandar, Can Liu, Andrew Y Ng, and Pranav Rajpurkar. Medaug: Contrastive learning leveraging patient metadata improves representations for chest x-ray interpretation. In *Machine Learning for Healthcare Conference*, pages 755–769, 2021.

[117] Bram Wallace and Bharath Hariharan. Extending and analyzing self-supervised learning across domains. In *European Conference on Computer Vision*, pages 717–734. Springer, 2020.

[118] Jiayu Wang, Wengang Zhou, Guo-Jun Qi, Zhongqian Fu, Qi Tian, and Houqiang Li. Transformation gan for unsupervised image synthesis and representation learning. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[119] Xiyue Wang, Sen Yang, Jun Zhang, Minghui Wang, Jing Zhang, Junzhou Huang, Wei Yang, and Xiao Han. Transpath: Transformer-based self-supervised learning for histopathological image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 186–195. Springer, 2021.

[120] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (csur)*, 53(3):1–34, 2020.

[121] Chen Wei, Lingxi Xie, Xutong Ren, Yingda Xia, Chi Su, Jiaying Liu, Qi Tian, and Alan L Yuille. Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 1910–1919, 2019.

[122] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Conference on Computer Vision and Pattern Recognition*, 2018.

[123] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *ICML*, 2021.

[124] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016.

[125] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Conference on Computer Vision and Pattern Recognition*, pages 1058–1067, 2017.

[126] Zongwei Zhou, Vatsal Sodha, Md Mahfuzur Rahman Siddiquee, Ruibin Feng, Nima Tajbakhsh, Michael B Gotway, and Jianming Liang. Models genesis: Generic autodidactic models for 3d medical image analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 384–393. Springer, 2019.