How Transferable are Self-supervised Features in Medical Image Classification Tasks?

Tuan Truong TUAN.TRUONG@BAYER.COM
Bayer AG, Germany

Sadegh Mohammadi SADEGH.MOHAMMADI@BAYER.COM
Bayer AG, Germany

Matthias Lenga MATTHIAS.LENGA@BAYER.COM
Bayer AG, Germany

Abstract

Transfer learning has become a standard practice to mitigate the lack of labeled data in medical classification tasks. Whereas finetuning a downstream task using supervised ImageNet pretrained features is straightforward and extensively investigated in many works, there is little study on the usefulness of self-supervised pretraining. This paper assesses the transferability of the most recent self-supervised ImageNet models, including SimCLR, SwAV, and DINO, on selected medical imaging classification tasks. The chosen tasks cover tumor detection in sentinel axillary lymph node images, diabetic retinopathy classification in fundus images, and multiple pathological condition classification in chest X-ray images. We demonstrate that self-supervised pretrained models yield richer embeddings than their supervised counterparts, benefiting downstream tasks for linear evaluation and finetuning. For example, at a critically small subset of the data with linear evaluation, we see an im provement up to 14.79% in Kappa score in the diabetic retinopathy classification task, 5.4% in AUC in the tumor classification task, 7.03% AUC in the pneumonia detection, and 9.4% in AUC in the detection of pathological conditions in chest X-ray. In addition, we introduce *Dynamic Visual Meta-Embedding* (DVME) as an end-to-end transfer learning approach that fuses pretrained embeddings from multiple models. We show that the collective representation obtained by DVME leads to a significant improvement in the performance of selected tasks compared to using a single pretrained model approach and can be generalized to any combination of pretrained models.

Keywords: Self-supervised learning, Transfer learning, Medical imaging

1. Introduction

1.1. Background and Motivation

The scarcity of high-quality annotated data remains a notorious challenge in medical image analysis due to the high cost of acquiring expert annotations (Castro et al., 2020). Transfer learning from large models pretrained in a supervised fashion on natural images such as ImageNet has become a *de-facto* solution for 2D medical imaging tasks in low data regimes (Lam et al., 2018; Bayramoglu and Heikkilä, 2016; Pardamean et al., 2018; Yang et al., 2018). Recently, self-supervised learning shows initial success in building large-scale Deep Learning based applications by leveraging unannotated data for pretraining (Zhou et al., 2019; Chen et al., 2019; Taleb et al., 2020, 2021; Zhuang et al., 2019; Bai et al., 2019; Abbet et al., 2020; Sowrirajan et al., 2021). However, a bottleneck within self-supervised learning is the demanding requirement of computational resources to train compared to standard supervised learning (Chen et al., 2020; Caron et al., 2021a,b; Grill et al., 2020). For example, regarding training on ImageNet, SwAV (Caron et al., 2021a) uses the batch size of 4096 distributed on 64 GPUs and SimCLR (Chen et al., 2020) uses varying batch sizes between 256 and 8192 on 32-128 TPU cores. Even when the batch size is small, the author of SimCLR notes that the training time must be extended to provide more negative examples. In pretraining medical datasets, Azizi et al. (2021) observe the best performance when using the batch size of 1024 on 64 cloud TPU cores to train Sim-

CLR on a chest X-ray dataset. While transfer learning from supervised pretraining on a large labeled dataset such as ImageNet is widely studied (Raghu et al., 2019; Ke et al., 2021), the transferability of models pretrained on ImageNet using self-supervised techniques requires further investigation.

This paper reflects on the effectiveness of transfer learning with self-supervised features. We evaluate the performance of four downstream classification tasks using ImageNet pretrained features obtained from supervised and self-supervised techniques. The four distinct tasks concern three modalities with varying data sizes and distributions. The first task is in the domain of digital pathology and aims to detect sentinel axillary lymph node metastases in hematoxylin and eosin (H&E) stained patches extracted from whole-slide images. The second task concerns the severity classification of diabetic retinopathy from colored fundus images. The last two tasks are related to reading X-ray images. One involves identifying whether a patient is suffering from pneumonia and the other involves detecting multiple findings, such as pneumothorax, nodule or mass, opacity, and fracture. In particular, we consider low data regimes ranging from approximately 1% to 10% of the original dataset size for each task (Section 5.3). We evaluate pretrained features of three self-supervised approaches, SimCLR (Chen et al., 2020), SwAV (Caron et al., 2021a), and DINO (Caron et al., 2021b), on aforementioned tasks by training a linear layer on top of frozen features. We find that DINO consistently outperforms other self-supervised techniques and the supervised baseline by a significant margin.

Additionally, we propose *Dynamic Visual Meta-Embeddings* (DVME) - a model-agnostic technique to combine multiple self-supervised pretrained features for downstream tasks. In natural language processing, it has been observed that different word embeddings work well for different tasks and that it is difficult to anticipate the usefulness of a given embedding technique for a certain task at hand. The usage of a meta-embedding mitigates this problem by constructing an ensemble of embedding sets to increase the lexical coverage of vocabulary which leads to improved performance on downstream tasks (Kiela et al., 2018). Similarly, in vision tasks, we propose to concatenate multiple pretrained embeddings with self-attention for transfer learning. Concatenation expands the embedding space and yields richer representation while self-attention adapts the contribution of individual embedding to a specific downstream task. We show that DVME leads to a further increase in performance across all tasks compared to the best single self-supervised pretrained model baseline.

1.2. Contributions

Overall, the main contributions are as following:

- Across four distinct medical image classification tasks, we assess the quality of the embeddings obtained from different models which are pretrained on ImageNet using state-of-the-art self-supervised or supervised pretraining techniques.
- We identify a single self-supervised model which consistently outperforms the other approaches on all selected downstream tasks. In particular, this effect is prominently observed in low data regimes.
- We propose Dynamic Visual Meta-Embeddings (DVME) to fully leverage the collective representations obtained from different self-supervised pretrained models. The representations obtained from the DVME model aggregation outperform all single model approaches on the selected downstream tasks.

2. Related work

Self-supervised learning in medical imaging Two main self-supervised approaches in medical imaging are in the form of *handcrafted pretext tasks* and *contrastive learning*. Early applications design tailored pretext tasks to reconstruct images from transformed or distorted inputs (Zhou et al., 2019; Chen et al., 2019; Taleb et al., 2021; Zhuang et al., 2019; Bai et al., 2019; Rivail et al., 2019; Abbet et al., 2020). For example, Model Genesis (Zhou et al., 2019) applies in-domain transfer learning to various classification and segmentation tasks on CT and X-ray images. The proposed architecture is an autoencoder that reconstructs images from four transformations: non-linear, local-shuffling, out-painting, and in-painting. The induced transformations are supposed to enable the encoder to learn features related to appearance, texture, and context. Chen et al. (2019) propose context restoration as a pretext task applied in three common medical use cases: plane detection on fetal 2D ultrasound images, abdominal organ localization on CT images, and brain tumor segmentation on MRI images. The proposed method

generates distorted images with different spatial contexts while maintaining the same intensity distribution by repeatedly swapping two random patches in an input image. Through reconstruction, the model learns useful semantic features transferable in subsequent target classification and segmentation tasks. In a different approach, Taleb et al. (2021) propose the multimodal puzzle task, inspired from the Jigsaw puzzles, which facilitates rich representation learning from multiple medical image modalities. The limitation of handcrafted pretext tasks is that they are highly task- and domain-specific, and thus cannot generalize well to different tasks. Lately, contrastive learning-based techniques (see Section 3) resolve this issue. Sowrirajan et al. (2021) use MoCo (He et al., 2020) to pretrain on unlabeled CheXpert (Irvin et al., 2019) dataset and finetunes with labels on external Shenzhen Hospital X-ray dataset (Jaeger et al., 2014) to detect pleural effusion. Dippel et al. (2021) extends a contrastive loss to a self-reconstruction task with attention mechanism on fundus images.

Transfer learning in medical imaging Transfer learning with ImageNet pretrained features still incites debates over its actual benefits for downstream medical tasks (Raghu et al., 2019; Ke et al., 2021; He et al., 2019). In a large data regime, Raghu et al. (2019) show that lightweight models with random initialization can perform on par with large architectures pretrained on ImageNet such as ResNet-50 (He et al., 2016) and Inception-v3 (Szegedy et al., 2015). On the contrary, Ke et al. (2021) argue that ImageNet pretraining can significantly boost the performance with newer architectures such as DenseNet (Huang et al., 2017) and EfficientNet (Tan and Le, 2019). In low data regimes, however, transfer learning with self-supervised approaches has been found particularly helpful in recent works (Newell and Deng, 2020; Azizi et al., 2021; Chaves et al., 2021). Azizi et al. (2021) perform transfer learning with SimCLR (Chen et al., 2020) on X-ray and dermatology datasets and show a significant gain compared to a supervised baseline. Chaves et al. (2021) evaluate self-supervised models on multiple dermatology datasets and find the advantage of self-supervised pretraining when using low training data. Whereas prior works focus on a single self-supervised technique (Azizi et al., 2021) and a unique modality, i.e., dermatology (Chaves et al., 2021), our work extends the investigation by benchmarking various self-supervised approaches against the supervised baseline across a set of heterogeneous medical imaging tasks. Our primary goal is to compare the richness of feature embeddings of different self-supervised learning techniques pretrained on ImgaeNet in the scope of transfer learning on medical imaging classification tasks.

3. Self-supervised Learning Techniques

An important line of work in self-supervised learning is contrastive learning where the representation is learned by comparing the similarity between images. The output embeddings obtained from an encoder are either pulled closer (similar) or pushed away (dissimilar) in the embedding space. Most of the contrastive approaches are built on the notion of *multi-instance level classification* (Dosovitskiy et al., 2016), which considers each image as a unique class and the model learns to discriminate it from the rest of the images in the batch. SimCLR and SwAV can be categorized into the group of contrastive learning. A detailed review and taxonomy of contrastive learning can be found in (Le-Khac et al., 2020). There is also another line of work which does not discriminate the instance but matches the output features with those from a momentum encoder. BYOL Grill et al. (2020) and DINO are examples from this line. The techniques of our focus in the paper are SimCLR, SwAV, and DINO.

SimCLR *Simple Framework for Contrastive Learning of Visual Representation* (Chen et al., 2020) maximizes the agreement of two views from the same image. The paper proposes a set of transformations applied to input images to create positive and negative pairs. An encoder takes a transformed batch and forwards it to a projection head that maps images to an embedding space. A contrastive loss on top compares the embeddings to minimize the distance between similar (positive) embeddings. Finally, the projection head is discarded and the encoder can be transferred to downstream tasks.

SwAV *Swapping Assignments between multiple Views of the same image* (Caron et al., 2021a) also contrasts two image views but not in a direct, sample-based fashion as SimCLR. Instead, it compares the cluster to which each view belongs. If two views come from the same image, they should fall on the same cluster assignment and vice versa. Caron et al. (2021a) show that this approach has an advantage over SimCLR in avoiding the need for large batch size and improving the convergence time. In comparison

to a prior clustering-based self-supervised technique in (Caron et al., 2018), the clustering assignment process is online, so that gradients can be backpropagated in a batch-wise manner.

DINO *Knowledge distillation without labels* (Caron et al., 2021b) matches the output probability distributions of two image views obtained from two networks. This approach takes inspiration from Bootstrap Your Own Latent (BYOL) (Grill et al., 2020) in the perspective of self distillation task and the architecture of Vision Transformer (ViT) (Dosovitskiy et al., 2021) as the backbone. Instead of passing the views into the same network, DINO passes two transformations of an image into two networks, namely the student and teacher network. The loss compares the probability outputs of both networks and the student's parameters are updated via backpropagation while the teacher's parameters is updated via an exponential moving average of the student ones. In addition, compared to using convolutional architectures, Caron et al. (2021b)'s study indicates that ViT-based DINO shows distinct properties in characterizing object boundaries and generates features that perform well using K-Nearest Neighbors without further finetuning in ImageNet classification task.

4. Datasets

The four datasets in our experiments are distinct in terms of modality, dataset size, and class distribution to partially reflect the heterogeneity of typical medical imaging tasks. We consider three common modalities in medical image analysis: digital pathology, fundus imaging, and X-ray.

PatchCamelyon (PatchCam) The dataset contains H&E sentinel axillary lymph node patches extracted from the whole-slide image in the study at (Ehteshami Bejnordi et al., 2017; Veeling et al., 2018). 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 positive. The data version we use is the curated one from Kaggle competition[1] that removes all the duplicated patches and comes with a default train/test split. The original train set consists of 220025 patches of size 96×96 with binary labels indicating whether there is a tumor or not. For our training task, we randomly select a subset comprising 50000 images. The official test set comprises 57486 images for which no labels are provided. Hence, for all performance evaluations on PatchCam, we submit our predictions to Kaggle.

APTOS The dataset comprises colored fundus images of Diabetic Retinopathy (DR) patients obtained from diverse clinics with different camera setups. For each image, clinicians rate the severity with a score between 0 and 4, indicating No-DR, Mild, Moderate, Severe, and Proliferative DR, respectively. The dataset is part of the challenge held on Kaggle[2]. The train and test sets contain 3662 and 1928 images, respectively. Similar to PatchCam, we submit the inference of the test set to Kaggle to obtain the scores.

Pneumonina chest X-ray The dataset contains chest X-ray images annotated by two expert radiologists. Each radiologist classifies each image into healthy and pneumonia. We obtain the dataset in Kaggle [3] with a default train/test split of 5216/624 images. Each patient can have multiple images, which is considered for patient stratification during training and validation. Further description of the dataset and acquisition can be found at (Kermany et al., 2018).

NIH chest X-ray The dataset consists of chest X-ray images provided by NIH Clinical Center[4]. Three certified radiologists manually reviewed the images. Each radiologist marks the presence of four medical conditions: pneumothorax, nodule or mass, opacity, and fracture. We use two subsets of the original NIH Chest X-ray dataset, which are referred to as validation and test set in the study at (Majkowska et al., 2020). We use the first subset (2414 images) for training and the second subset (1962 images) for evaluation. Since there can be multiple findings per image, we exclude such cases in our experiment for simplicity. In addition, we also add a class called *other* for when no mentioned conditions are found in the image.

5. Experimental Setup

5.1. Architecture

To assess the transferability of self-supervised compare to fully supervised features pretrained on Ima-

1. https://www.kaggle.com/c/histopathologic-cancer-detection
2. https://www.kaggle.com/c/aptos2019-blindness-detection
3. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
4. https://nihcc.app.box.com/v/ChestXray-NIHCC

geNet, we used ResNet-50 He et al. (2016), as it is the backbone for state-of-the-art self-supervised approaches (Chen et al., 2020; He et al., 2020; Caron et al., 2021a), including SwAV (Caron et al., 2021a) and SimCLR (Chen et al., 2020). On the other hand, for DINO (Caron et al., 2021b), the architecture is VisionTransformer (ViT) with patch size 8×8. We rely on a well-established computer VIsion library for state-of-the-art Self-Supervised Learning (VISSL) (Goyal et al., 2021) for SwAV and SimCLR. At the same time, for DINO, we used available pretrained weight on the Facebook research repository [5].

5.2. Hyperparameters and Augmentation

All experiments use the Adam optimizer starting with a small learning rate between 1e-3 and 1e-4 and further reducing it when the validation loss does not improve consecutively over five epochs. As our study aims to compare different initializations and not outperform the best performance, we do not apply intensive augmentation techniques. Images with an original size larger than 224×224 are resized into 256×256, then cropped and applied further flipping or rotation depending on the nature of modality. For the PatchCam dataset, we apply directly flipping without resizing or cropping as the size of the images is less than 224×224.

5.3. Dataset sizes and subtasks

To fully assess the transferability of self-supervised features under various data regimes, we define three different subtasks, Small (S), Medium (M), and Full (F). Table 1 reports detailed information regarding data splits for each dataset. We construct the subtasks with a five-fold cross-validation fashion. We randomly extract samples from the entire dataset, conduct the training and validation for each split, and repeat this process five times. Then, we select the best performing model on the validation set to the fixed training set. To avoid bias learning due to class imbalance, we maintain the number of samples per class balance. The only exception is the NIH Chest X-ray dataset; we used an oversampling strategy during the training procedure because of a significant class imbalance.

5.4. Metrics

The evaluation metric for the PatchCam, Pneumonia and NIH Chest X-ray is the area under the Receiver Operating Characteristic curve (AUC) while the Cohen Kappa score for APTOS. We submit the APTOS and PatchCam test set predictions to Kaggle and obtain two scores for the private and public leaderboards, which are evaluated on two different portions of the test set. We calculate the final score as the weighted average score of the private and public leaderboard. Precisely, the final score is calculated as $s_{avg} = \alpha \times s_{private} + (1 - \alpha) \times s_{public}$ where α for PatchCam is 0.51 and for APTOS is 0.85. The value of α is the portion of the test set that Kaggle uses to calculate the private leaderboard score and varies depending on the competition.

Table 1: Number of samples for different subtasks

Dataset	S	M	F	Test
PatchCam	500	5000	50000	57486
APTOS	50	500	3662	1928
Pneumonia Chest X-ray	50	500	5216	624
NIH Chest X-ray	20	200	2414	1962

5.5. Linear performance and finetuning

We conduct comprehensive experiments on pretrained models on ImageNet by two experiments: *linear evaluation* on frozen features and *finetuning* with labels of downstream tasks. For SwAV and SimCLR and fully supervised pretrained models on ImageNet, we add the linear layer after the last average pooling layer in ResNet-50. At the same time, for DINO, we follow the implementation of Caron et al. (2021b) to add a linear layer after the concatenation of class tokens from the last four blocks in ViT. In addition to linear evaluation we consider *finetuning*, where all layers of the pretrained base network as well as the final linear classifier are adapted on the downstream task at hand.

5.6. Linear evaluation with DVME

Given a set of pretrained feature extractors, it may be difficult to anticipate which pretrained model to choose for a given downstream task at hand. This concern is also shared in natural language processing where there are multiple word embedding techniques trained on different domains, each having its own strengths depending on target tasks. Meta-embedding is an effective technique that takes a union over different word embeddings to tackle the out-of-vocabulary problem and fuse multi-modal informa-

5. https://github.com/facebookresearch/dino

tion (Kiela et al., 2018). Though there is no multimodal information in our study, we hypothesize that the pretrained features from different techniques are sufficiently independent of one another and encode certain complementary information. Therefore, we propose Dynamic Visual Meta-Embedding (DVME) for vision tasks, aggregating information by concatenating the embeddings of SimCLR, SwAV, and DINO pretrained models. The newly constructed embedding space improves the separability of image features through the complementary effect of each embedding component. We extract the embedding space from the last fully connected unit from SimCLR and SwAV with the dimension 2048. For DINO, we construct the embedding by concatenating the class token of the last four blocks results in the dimension of 1536. Then, we project each embedding into a fixed size of 512 and feed the concatenation of the resulting embedding into a self-attention module. Then, we concatenate the embedding space and project it to a fixed dimension of 512 to learn the importance of each embedding component for a specific downstream task. The self-attention module is the same as in the Vision Transformer architecture (Dosovitskiy et al., 2021) except that the attention is learned across different components of the meta-embedding instead of image patches. Figure 1 shows a sketch of how self-attention is incorporated when fusing the pretrained features. The proposed DVME approach is not limited to SimCLR, SwAV, or DINO and can be used with other feature extractors. We provide a snippet of DVME implemented in PyTorch in Appendix Section E.

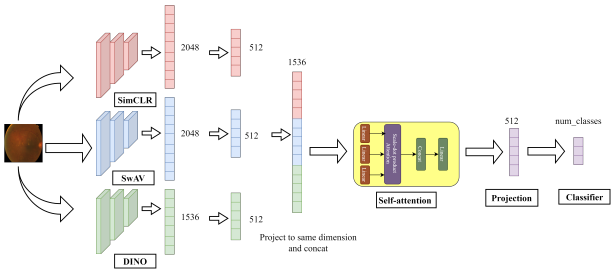


Figure 1: Dynamic Visual Meta-Embedding (DVME): The embeddings extracted from each pretrained model are projected to the same dimension and concatenated before feeding to the self-attention module.

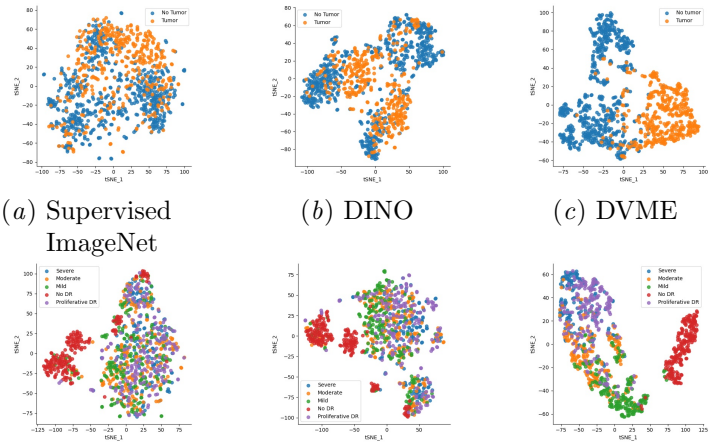
6. Results and Discussion

6.1. Evaluation of self-supervised and supervised pretrained features

We test the generalization of self-supervised and fully supervised pretrained features on ImageNet by transferring them to several downstream medical imaging classification tasks under various data regimes. Table 2 shows *Linear evaluation* methods across various datasets. It is visible that SwAV and SimCLR pretrained features yield inconsistent patterns across all downstream tasks. For example, while SwAV and SimCLR initializations perform on par with each other on PatchCam and NIH Chest X-ray, they are different by approximately 10% in Kappa score and 3.7% in AUC on the S subsets of APTOS and Pneumonia Chest X-ray, respectively. Notably, DINO initialization consistently outperforms all the other initializations across all tasks by a significant margin. For instance, on NIH Chest X-ray S and M subtasks, DINO pretrained features improve approximately 5-6% in AUC over SimCLR and SwAV. The single exception is the APTOS S subtask, where SwAV outperforms DINO by 3.3% in AUC. However, DINO still yields an improvement over SimCLR and ImageNet supervised initialization by 7% and 11.2% in Kappa score, respectively. We refer to Appendix B.1 for more detailed results on the performance obtained by the competing initialization methods for different dataset sizes. In comparison to ImageNet supervised pretrained features, we observe that self-supervised features improve the performance across all downstream tasks. This suggests that the representation generated by self-supervised methods are of higher quality, leading to better performance on the test set and reducing the performance variability between folds in low data regimes, similar to the observation made in Chaves et al. (2021).

Figure 2 (a,b,d,e) shows the t-SNE visualization of the features extracted from the supervised pretrained ResNet-50 and DINO on the PatchCam (binary classification) and APTOS (multi-class) downstream tasks. It is visible that DINO offers a clear class separation compare to its supervised counterpart. We observe the same behavior for other datasets, which we refer to Appendix D for further detail.

We extend our comparison by *finetuning* model initializations separately on all downstream tasks. Table 3 summarizes the finetuning results across datasets and their subtasks. Similar to the linear



(a) Supervised ImageNet (b) DINO (c) DVME

(d) Supervised ImageNet (e) DINO (f) DVME

Figure 2: t-SNE visualization of embeddings obtained using different pretrained feature extractors (supervised ImageNet, DINO, proposed method DVME). Top row: **PatchCam** dataset, bottom row: **APTOS** dataset

Table 2: Linear evaluation performance of different self-supervised initializations, supervised pretraining and random initialization on different scales (small, medium, full) of the data.

Dataset	Method	Small (S)	Medium (M)	Full (F)
PatchCam	Random	0.6594 ± 0.0319	0.6994 ± 0.0079	0.7990 ± 0.0021
	Supervised ImageNet	0.7517 ± 0.0136	0.7863 ± 0.0063	0.7975 ± 0.0032
	SwAV	0.7834 ± 0.0112	0.8043 ± 0.0072	0.8088 ± 0.0025
	SimCLR	0.7895 ± 0.0091	0.8053 ± 0.0069	0.8084 ± 0.0026
	DINO	**0.8058 ± 0.0100**	**0.8359 ± 0.0053**	**0.8487 ± 0.0014**
APTOS (*)	Random	0.0324 ± 0.0602	0.0624 ± 0.0459	0.1550 ± 0.1160
	Supervised ImageNet	0.4851 ± 0.0811	0.6822 ± 0.0257	0.7331 ± 0.0124
	SwAV	**0.6330 ± 0.0204**	0.7274 ± 0.0095	0.7617 ± 0.0128
	SimCLR	0.5305 ± 0.0539	0.6500 ± 0.0138	0.6989 ± 0.0084
	DINO	0.6003 ± 0.0691	**0.7372 ± 0.0167**	**0.7790 ± 0.0083**
Pneumonia Chest X-ray	Random	0.6899 ± 0.0339	0.8258 ± 0.0237	0.8907 ± 0.0144
	Supervised ImageNet	0.8789 ± 0.0234	0.8954 ± 0.0151	0.9397 ± 0.0033
	SwAV	0.8808 ± 0.0222	0.9215 ± 0.0252	0.9709 ± 0.0047
	SimCLR	0.9168 ± 0.0006	0.9346 ± 0.0072	0.9665 ± 0.0027
	DINO	**0.9492 ± 0.0170**	**0.9718 ± 0.0055**	**0.9868 ± 0.0008**
NIH Chest X-ray	Random	0.5212 ± 0.0344	0.5317 ± 0.0176	0.5392 ± 0.0346
	Supervised ImageNet	0.5383 ± 0.0392	0.6688 ± 0.0148	0.7109 ± 0.0084
	SwAV	0.5785 ± 0.0258	0.6889 ± 0.0089	0.7225 ± 0.0139
	SimCLR	0.5792 ± 0.0435	0.6645 ± 0.0067	0.6983 ± 0.0231
	DINO	**0.6323 ± 0.0131**	**0.7373 ± 0.0112**	**0.7438 ± 0.0228**

(*) The evaluation metric for APTOS is Cohen-Kappa score while for others is AUC score.

Table 3: Finetuning performance of different self-supervised initializations, supervised pretraining and random initialization on different scales (small, medium, full) of the data.

Dataset	Method	Small (S)	Medium (M)	Full (F)
PatchCam	Random	0.7355 ± 0.0282	0.7660 ± 0.0223	0.8515 ± 0.0023
	Supervised ImageNet	0.7897 ± 0.0162	0.8274 ± 0.0051	0.8483 ± 0.0097
	SwAV	0.7895 ± 0.0336	0.8399 ± 0.0142	**0.8619 ± 0.0090**
	SimCLR	0.8021 ± 0.0138	0.8329 ± 0.0085	0.8553 ± 0.0110
	DINO	**0.8366 ± 0.0092**	**0.8440 ± 0.0172**	0.8517 ± 0.0158
APTOS (*)	Random	0.0177 ± 0.0954	0.3233 ± 0.0822	0.5927 ± 0.0545
	Supervised ImageNet	0.4817 ± 0.0991	0.7369 ± 0.0310	0.8057 ± 0.0149
	SwAV	0.4928 ± 0.0378	0.7594 ± 0.0246	0.8293 ± 0.0133
	SimCLR	0.5916 ± 0.0570	0.7603 ± 0.0249	0.8264 ± 0.0103
	DINO	**0.6601 ± 0.0447**	**0.7945 ± 0.0079**	**0.8365 ± 0.0213**
Pneumonia Chest X-ray	Random	0.6895 ± 0.0512	0.9183 ± 0.0186	0.9820 ± 0.0043
	Supervised ImageNet	0.8649 ± 0.0442	0.9698 ± 0.0066	0.9910 ± 0.0015
	SwAV	**0.9289 ± 0.0291**	0.9814 ± 0.0087	0.9927 ± 0.0016
	SimCLR	0.9197 ± 0.0168	0.9781 ± 0.0085	**0.9950 ± 0.0013**
	DINO	0.9256 ± 0.0235	**0.9867 ± 0.0051**	0.9948 ± 0.0010
NIH Chest X-ray	Random	0.5015 ± 0.0253	0.6404 ± 0.0165	0.6616 ± 0.0345
	Supervised ImageNet	0.5251 ± 0.0238	0.6816 ± 0.0429	0.7618 ± 0.0116
	SwAV	**0.5903 ± 0.0384**	0.6973 ± 0.0227	**0.7737 ± 0.0212**
	SimCLR	0.5570 ± 0.0450	**0.7228 ± 0.0287**	0.7358 ± 0.0295
	DINO	0.5552 ± 0.0546	0.6652 ± 0.0114	0.7404 ± 0.0240

(*) The evaluation metric for APTOS is Cohen-Kappa score while for others is AUC score.

Table 4: Linear evaluation performance of Dynamic Visual Meta-Embedding (DVME) in comparison with the best score obtained using a single pretrained model on different downstream tasks on different scales (small, medium, full) of the data.

Dataset	Method	Small (S)	Medium (M)	Full (F)
PatchCam	DVEM	**0.8227 ± 0.0148**	**0.8399 ± 0.0059**	0.8467 ± 0.0094
	Best single baseline	0.8058 ± 0.0100	0.8359 ± 0.0100	**0.8487 ± 0.0014**
APTOS (*)	DVME	**0.6913 ± 0.0575**	**0.7925 ± 0.0265**	**0.8242 ± 0.0279**
	Best single baseline	0.6330 ± 0.0204	0.7372 ± 0.0167	0.7790 ± 0.0083
Pneumonia Chest X-ray	DVME	**0.9539 ± 0.0025**	0.9696 ± 0.0101	0.9842 ± 0.0029
	Best single baseline	0.9492 ± 0.0170	**0.9718 ± 0.0055**	**0.9868 ± 0.0008**
NIH Chest X-ray	DVME	**0.6566 ± 0.0564**	**0.7601 ± 0.0146**	**0.7538 ± 0.0234**
	Best single baseline	0.6323 ± 0.0131	0.7373 ± 0.0112	0.7438 ± 0.0228

(*) The evaluation metric for APTOS is Cohen-Kappa score while for others is AUC score.

evaluation results, we consistently observe a higher performance for all self-supervised pretrained initializations compared to the supervised pretrained and randomly initialized baselines in the low data regimes (S, M), which supports the observation made by Azizi et al. (2021); Chaves et al. (2021). DINO pretrained features outperform those from other self-supervised methods in 2/4 S subtasks and 3/4 M subtasks. When using full data for fine-tuning, all self-supervised pretrained initializations consistently outperform the baseline methods on PatchCam, APTOS and Pneumonia Chest X-ray. However, for full NIH Chest X-ray task data, only SwAV exceeds the supervised baseline performance.

Moreover, we observe that for S subtask of APTOS, Pneumonia Chest X-ray, and NIH Chest X-ray dataset where there are 50 samples or less finetuning leads to overfitting. For example, while DINO achieves the highest AUC of 0.6323 on the S subtask of NIH Chest X-ray in linear evaluation, the best performance for finetuning is obtained by SwAV and reaches only 0.5903 AUC. In the APTOS dataset, SwAV initialization reaches 0.6330 at the small dataset size in linear evaluation but drops to 0.4928 in finetuning. However, when the number of samples is increased up to a few thousand, the finetuning performance is higher than linear evaluation across all of the tasks. When using full data, the best performance using finetuning mounts up to 1.32% in AUC, 5.75% in Kappa score, 0.85% in AUC, and 3% in AUC in PatchCam, APTOS, Pneumonia Chest X-ray, and NIH Chest X-ray tasks, respectively. The improvement can be attributed to the increase in training samples, helping the model fit well to the downstream data but maintaining a decent generalization.

6.2. Evaluation of DVME performance

Table 4 summarizes the results obtained from fusing SwAV, SimCLR, and DINO with the DVME approach. We evaluate DVME similar to the linear performance evaluation from Section 6.1 by training only the meta-embedding on top of the three frozen feature extractors, cf. Section 5.6. As a performance benchmark, we select the self-supervised initialization for each dataset and each fraction of data that leads to the best linear evaluation performance, cf. Table 2. DVME outperforms this benchmark in 4/4 of the S subtasks, 3/4 of the M subtasks, and 2/4 F subtasks. For the subtasks where DVME is not exceeding the benchmark performance, the difference lies within one standard deviation of the DVME linear evaluation score. The improvement of DVME over the benchmark is particularly pronounced for the APTOS and NIH Chest X-ray tasks. For example, DVME helps gain roughly 6% in Kappa score over the best individual baseline for the S and M subtask of the APTOS dataset.

The t-SNE visualizations of the DVME embeddings in Figure 2 (c,f) qualitatively indicate that the clusters are better separated, particularly in the case of multiclass classification. When analyzing the attention matrix, we find that SwAV and SimCLR pay little attention to each other but firmly into DINO, suggesting the representation from SwAV and SimCLR could be more similar and thus not so informative compared to DINO.

To better understand the effect of self-attention on the model fusion, we conduct an ablation study on DVME in Appendix C. In the setting without self-attention, the meta-embedding is directly connected to the linear classifier. Without self-attention, the feature fusion still yields a significant improvement over the baseline, which supports our hypothesis in Section 5.6 that each embedding contains complementary information. However, self-attention is particularly beneficial to specific tasks. For example, on APTOS, the Kappa scores increase by 5.6%, 4.4%, and 4.8% for the S, M, and F subtask, respectively.

7. Conclusion

This study assesses the quality of ImageNet self-supervised pretrained features in four selected medical image classification tasks. We demonstrate that feature extractor that is pretrained using SwAV, SimCLR, or DINO consistently yield richer embeddings on the downstream tasks than a supervised pretrained baseline model. Among all self-supervised techniques, DINO outperforms the other methods on the majority of datasets and subtasks. Furthermore, we show that each pretrained model's representations encode complementary information that can be fused to yield even more meaningful features. Therefore, we propose Dynamic Visual Meta-Embedding (DVME), a model-agnostic meta-embedding approach. Our experiments indicate that DVME outperforms the best single model baseline on all downstream tasks. As a model-agnostic approach, DVME is not limited to SwAV, SimCLR, or DINO. With slight modifications, other models can be combined using DVME to generate enriched representations.

References

Christian Abbet, Inti Zlobec, Behzad Bozorgtabar, and Jean-Philippe Thiran. Divide-and-Rule: Self-Supervised Learning for Survival Analysis in Colorectal Cancer. In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, volume 12265, pages 480–489. Springer International Publishing, Cham, 2020. ISBN 978-3-030-59721-4 978-3-030-59722-1. doi: 10.1007/978-3-030-59722-1_46. URL http://link.springer.com/10.1007/978-3-030-59722-1_46. Series Title: Lecture Notes in Computer Science.

Shekoofeh Azizi, Basil Mustafa, Fiona Ryan, Zachary Beaver, Jan Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, Ting Chen, Vivek Natarajan, and Mohammad Norouzi. Big Self-Supervised Models Advance Medical Image Classification. *arXiv:2101.05224 [cs, eess]*, January 2021. URL http://arxiv.org/abs/2101.05224. arXiv: 2101.05224.

Wenjia Bai, Chen Chen, Giacomo Tarroni, Jinming Duan, Florian Guitton, Steffen E. Petersen, Yike Guo, Paul M. Matthews, and Daniel Rueckert. Self-Supervised Learning for Cardiac MR Image Segmentation by Anatomical Position Prediction. In Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap, and Ali Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, volume 11765, pages 541–549. Springer International Publishing, Cham, 2019. ISBN 978-3-030-32244-1 978-3-030-32245-8. doi: 10.1007/978-3-030-32245-8_60. URL http://link.springer.com/10.1007/978-3-030-32245-8_60. Series Title: Lecture Notes in Computer Science.

Neslihan Bayramoglu and Janne Heikkilä. Transfer Learning for Cell Nuclei Classification in Histopathology Images. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, volume 9915, pages 532–539. Springer International Publishing, Cham, 2016. ISBN 978-3-319-49408-1 978-3-319-49409-8. doi: 10.1007/978-3-319-49409-8_46. URL http://link.springer.com/10.1007/978-3-319-49409-8_46. Series Title: Lecture Notes in Computer Science.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep Clustering for Unsupervised Learning of Visual Features. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11218, pages 139–156. Springer International Publishing, Cham, 2018. ISBN 978-3-030-01263-2 978-3-030-01264-9. doi: 10.1007/978-3-030-01264-9_9. URL http://link.springer.com/10.1007/978-3-030-01264-9_9. Series Title: Lecture Notes in Computer Science.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. *arXiv:2006.09882 [cs]*, January 2021a. URL http://arxiv.org/abs/2006.09882. arXiv: 2006.09882.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. *arXiv:2104.14294 [cs]*, April 2021b. URL http://arxiv.org/abs/2104.14294. arXiv: 2104.14294.

Daniel C. Castro, Ian Walker, and Ben Glocker. Causality matters in medical imaging. *Nature Communications*, 11(1):3673, December 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-17478-w. URL http://www.nature.com/articles/s41467-020-17478-w.

Levy Chaves, Alceu Bissoto, Eduardo Valle, and Sandra Avila. An Evaluation of Self-Supervised Pre-Training for Skin-Lesion Analysis. *arXiv:2106.09229 [cs]*, June 2021. URL http://arxiv.org/abs/2106.09229. arXiv: 2106.09229.

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, December 2019. ISSN 13618415. doi: 10.1016/j.media.2019.101539. URL https://linkinghub.elsevier.com/retrieve/pii/S1361841518304699.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *In-*

ternational conference on machine learning, pages 1597–1607. PMLR, 2020.

Jonas Dippel, Steffen Vogler, and Johannes Höhne. Towards fine-grained visual representations by combining contrastive learning with image reconstruction and attention-weighted pooling. *arXiv preprint arXiv:2104.04323*, 2021.

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, September 2016. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2015.2496141. URL https://ieeexplore.ieee.org/document/7312476/.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]*, June 2021. URL http://arxiv.org/abs/2010.11929. arXiv: 2010.11929.

Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, , and the CAMELYON16 Consortium. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA*, 318(22): 2199–2210, 12 2017. ISSN 0098-7484. doi: 10.1001/jama.2017.14585. URL https://doi.org/10.1001/jama.2017.14585.

Priya Goyal, Quentin Duval, Jeremy Reizenstein, Matthew Leavitt, Min Xu, Benjamin Lefaudeux, Mannat Singh, Vinicius Reis, Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Ishan Misra. Vissl. https://github.com/facebookresearch/vissl, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised Learning. *arXiv:2006.07733 [cs, stat]*, September 2020. URL http://arxiv.org/abs/2006.07733. arXiv: 2006.07733.

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

Kaiming He, Ross Girshick, and Piotr Dollar. Rethinking ImageNet Pre-Training. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4917–4926, Seoul, Korea (South), October 2019. IEEE. ISBN 978-1-72814-803-8. doi: 10.1109/ICCV.2019.00502. URL https://ieeexplore.ieee.org/document/9010930/.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, Seattle, WA, USA, June 2020. IEEE. ISBN 978-1-72817-168-5. doi: 10.1109/CVPR42600.2020.00975. URL https://ieeexplore.ieee.org/document/9157636/.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, Jayne Seekins, David A. Mong, Safwan S. Halabi, Jesse K. Sandberg, Ricky Jones, David B. Larson, Curtis P. Langlotz, Bhavik N. Patel, Matthew P. Lungren, and Andrew Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison, 2019.

Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J. Wáng, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative Imaging in Medicine and Surgery*, 4(6), 2014. ISSN 2223-4306. URL https://qims.amegroups.com/article/view/5132.

Alexander Ke, William Ellsworth, Oishi Banerjee, Andrew Y. Ng, and Pranav Rajpurkar. CheXtransfer: Performance and Parameter Efficiency of ImageNet Models for Chest X-Ray Interpretation. *Proceedings of the Conference on Health, Inference, and Learning*, pages 116–124, April 2021. doi: 10.1145/3450439.3451867. URL http://arxiv.org/abs/2101.06871. arXiv: 2101.06871.

Daniel S. Kermany, Michael Goldbaum, Wenjia Cai, Carolina C.S. Valentim, Huiying Liang, Sally L. Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, Justin Dong, Made K. Prasadha, Jacqueline Pei, Magdalene Y.L. Ting, Jie Zhu, Christina Li, Sierra Hewett, Jason Dong, Ian Ziyar, Alexander Shi, Runze Zhang, Lianghong Zheng, Rui Hou, William Shi, Xin Fu, Yaou Duan, Viet A.N. Huu, Cindy Wen, Edward D. Zhang, Charlotte L. Zhang, Oulan Li, Xiaobo Wang, Michael A. Singer, Xiaodong Sun, Jie Xu, Ali Tafreshi, M. Anthony Lewis, Huimin Xia, and Kang Zhang. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5):1122–1131.e9, 2018. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2018.02.010. URL https://www.sciencedirect.com/science/article/pii/S0092867418301545.

Douwe Kiela, Changhan Wang, and Kyunghyun Cho. Dynamic Meta-Embeddings for Improved Sentence Representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1477, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1176. URL http://aclweb.org/anthology/D18-1176.

Carson Lam, Darvin Yi, Margaret Guo, and Tony Lindsey. Automated detection of diabetic retinopathy using deep learning. *AMIA summits on translational science proceedings*, 2018:147, 2018.

Phuc H. Le-Khac, Graham Healy, and Alan F. Smeaton. Contrastive Representation Learning: A Framework and Review. *IEEE Access*, 8: 193907–193934, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3031549. URL https://ieeexplore.ieee.org/document/9226466/.

Anna Majkowska, Sid Mittal, David F. Steiner, Joshua J. Reicher, Scott Mayer McKinney, Gavin E. Duggan, Krish Eswaran, Po-Hsuan Cameron Chen, Yun Liu, Sreenivasa Raju Kalidindi, Alexander Ding, Greg S. Corrado, Daniel Tse, and Shravya Shetty. Chest radiograph interpretation with deep learning models: Assessment with radiologist-adjudicated reference standards and population-adjusted evaluation. *Radiology*, 294(2):421–431, 2020. doi: 10.1148/radiol.2019191293. URL https://doi.org/10.1148/radiol.2019191293. PMID: 31793848.

Alejandro Newell and Jia Deng. How Useful Is Self-Supervised Pretraining for Visual Tasks? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7343–7352, Seattle, WA, USA, June 2020. IEEE. ISBN 978-1-72817-168-5. doi: 10.1109/CVPR42600.2020.00737. URL https://ieeexplore.ieee.org/document/9157100/.

Bens Pardamean, Tjeng Wawan Cenggoro, Reza Rahutomo, Arif Budiarto, and Ettikan Kandasamy Karuppiah. Transfer learning from chest x-ray pretrained convolutional neural network for learning mammogram data. *Procedia Computer Science*, 135:400–407, 2018.

Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding Transfer Learning for Medical Imaging. *arXiv:1902.07208 [cs, stat]*, October 2019. URL http://arxiv.org/abs/1902.07208. arXiv: 1902.07208.

Antoine Rivail, Ursula Schmidt-Erfurth, Wolf-Dieter Vogl, Sebastian M. Waldstein, Sophie Riedl, Christoph Grechenig, Zhichao Wu, and Hrvoje Bogunovic. Modeling Disease Progression in Retinal OCTs with Longitudinal Self-supervised Learning. In Islem Rekik, Ehsan Adeli, and Sang Hyun Park, editors, *Predictive Intelligence in Medicine*, volume 11843, pages 44–52. Springer International Publishing, Cham, 2019. ISBN 978-3-030-32280-9 978-3-030-32281-6. doi: 10.1007/978-3-030-32281-6_5. URL http://link.springer.com/10.1007/978-3-030-32281-6_5. Series Title: Lecture Notes in Computer Science.

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. PMLR, 2021.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015. doi: 10.1109/CVPR.2015.7298594.

Aiham Taleb, Winfried Loetzsch, Noel Danz, Julius Severin, Thomas Gaertner, Benjamin Bergner, and Christoph Lippert. 3D Self-Supervised Methods for Medical Imaging. *arXiv:2006.03829 [cs, eess]*, November 2020. URL http://arxiv.org/abs/2006.03829. arXiv: 2006.03829.

Aiham Taleb, Christoph Lippert, Tassilo Klein, and Moin Nabi. Multimodal self-supervised learning for medical image analysis. In *International Conference on Information Processing in Medical Imaging*, pages 661–673. Springer, 2021.

Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/tan19a.html.

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.

Yang Yang, Lin-Feng Yan, Xin Zhang, Yu Han, Hai-Yan Nan, Yu-Chuan Hu, Bo Hu, Song-Lin Yan, Jin Zhang, Dong-Liang Cheng, et al. Glioma grading on conventional mr images: a deep learning study with transfer learning. *Frontiers in neuroscience*, 12:804, 2018.

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.

Xinrui Zhuang, Yuexiang Li, Yifan Hu, Kai Ma, Yujiu Yang, and Yefeng Zheng. Self-supervised Feature Learning for 3D Medical Images by Playing a Rubik's Cube. In Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap, and Ali Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, volume 11767, pages 420–428. Springer International Publishing, Cham, 2019. ISBN 978-3-030-32250-2 978-3-030-32251-9. doi: 10.1007/978-3-030-32251-9_46. URL http://link.springer.com/10.1007/978-3-030-32251-9_46. Series Title: Lecture Notes in Computer Science.

Appendix A. Dataset splits

For each experiment on a single dataset, we split the dataset into 5 training and validation folds. Each training fold contains the same number of samples per each class. An exception is the NIH dataset since the number of samples across classes is highly imbalanced. In this case, we continue sampling to maximize the number of samples per each class as much as possible and use oversampling during the training process to compensate for class imbalance. In section B, we report the sample size in absolute values across all of our experiments.

Appendix B. Detailed results

B.1. Linear Evaluation

For all experiments in linear evaluation, we replace the last layer of the pretrained model with a new linear classifier and train only this layer. The minimum and maximum number of epochs that we train our models are 30 and 50 respectively. In addition, we set early stopping with the patience of 10 epochs. The initial learing rate is 0.001 and is reduced with a factor of 0.1 by the `ReduceLROnPlateau` scheduler when the validation score does not improve for 5 epochs consecutively. The batch size of 64 is kept fixed across all experiments.

Table B1: Linear evaluation on the PatchCam dataset with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50	0.5041 ± 0.0091	0.7193 ± 0.0199	0.7543 ± 0.0342	0.7502 ± 0.0376	0.7721 ± 0.0233
100	0.5001 ± 0.0002	0.7287 ± 0.0218	0.7756 ± 0.0280	0.7714 ± 0.0387	0.8040 ± 0.0050
200	0.5629 ± 0.0310	0.7183 ± 0.0224	0.7510 ± 0.0345	0.7675 ± 0.0189	0.8010 ± 0.0107
500 (S)	0.6594 ± 0.0319	0.7517 ± 0.0136	0.7834 ± 0.0112	0.7895 ± 0.0091	0.8058 ± 0.0100
1000	0.6886 ± 0.0090	0.7667 ± 0.0087	0.7686 ± 0.0509	0.7981 ± 0.0024	0.8204 ± 0.0106
2000	0.6955 ± 0.0168	0.7709 ± 0.0075	0.8022 ± 0.0071	0.7996 ± 0.0076	0.8214 ± 0.0116
5000 (M)	0.6994 ± 0.0079	0.7863 ± 0.0063	0.8043 ± 0.0072	0.8053 ± 0.0069	0.8359 ± 0.0053
10000	0.7110 ± 0.0046	0.7894 ± 0.0046	0.8338 ± 0.0163	0.8051 ± 0.0050	0.8399 ± 0.0029
20000	0.7210 ± 0.0081	0.7970 ± 0.0061	0.8310 ± 0.0356	0.8110 ± 0.0048	0.8446 ± 0.0033
Full	0.7990 ± 0.0021	0.7975 ± 0.0032	0.8088 ± 0.0025	0.8084 ± 0.0026	0.8487 ± 0.0014

Table B2: Linear evaluation on the APTOS dataset with various initializations. The mean Kappa score is obtained across 5 folds.

Mean Kappa Score					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50 (S)	0.0324 ± 0.0602	0.4851 ± 0.0811	0.6330 ± 0.0204	0.5305 ± 0.0539	0.6003 ± 0.0691
100	-0.0272 ± 0.0300	0.5758 ± 0.0435	0.6559 ± 0.0260	0.5657 ± 0.0611	0.6889 ± 0.0433
200	0.0083 ± 0.0429	0.6752 ± 0.0219	0.7100 ± 0.0118	0.6369 ± 0.0084	0.7339 ± 0.0244
500 (M)	0.0624 ± 0.0459	0.6822 ± 0.0257	0.7274 ± 0.0095	0.6500 ± 0.0138	0.7372 ± 0.0167
Full	0.1550 ± 0.1160	0.7331 ± 0.0124	0.7617 ± 0.0128	0.6989 ± 0.0084	0.7790 ± 0.0083

Table B3: Linear evaluation on the Pneumonia Chest X-Ray dataset with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50 (S)	0.6899 ± 0.0339	0.8789 ± 0.0234	0.8808 ± 0.0222	0.9168 ± 0.0006	0.9492 ± 0.0170
100	0.7323 ± 0.0602	0.8788 ± 0.0197	0.8731 ± 0.0260	0.9010 ± 0.0337	0.9466 ± 0.0154
200	0.7720 ± 0.0247	0.8789 ± 0.0315	0.8753 ± 0.0246	0.9176 ± 0.0154	0.9553 ± 0.0136
500 (M)	0.8258 ± 0.0237	0.8954 ± 0.0151	0.9215 ± 0.0252	0.9346 ± 0.0072	0.9718 ± 0.0055
Full	0.8907 ± 0.0144	0.9397 ± 0.0033	0.9709 ± 0.0047	0.9665 ± 0.0027	0.9868 ± 0.0008

Table B4: Linear evaluation on the NIH Chest X-ray dataset with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
20 (S)	0.5212 ± 0.0344	0.5383 ± 0.0392	0.5785 ± 0.0258	0.5792 ± 0.0435	0.6323 ± 0.0131
50	0.5127 ± 0.0172	0.5897 ± 0.0283	0.6469 ± 0.0140	0.6273 ± 0.0130	0.6831 ± 0.0233
100	0.5031 ± 0.0465	0.6388 ± 0.0169	0.6563 ± 0.0238	0.6359 ± 0.0375	0.6686 ± 0.0227
150	0.5044 ± 0.0216	0.6432 ± 0.0283	0.6673 ± 0.0272	0.6686 ± 0.0143	0.7385 ± 0.0243
200 (M)	0.5317 ± 0.0176	0.6688 ± 0.0148	0.6889 ± 0.0089	0.6645 ± 0.0067	0.7373 ± 0.0112
Full	0.5392 ± 0.0346	0.7109 ± 0.0084	0.7225 ± 0.0139	0.6983 ± 0.0231	0.7438 ± 0.0228

B.2. Finetuning

We keep all the hyperparameters the same as linear evaluation (Section B.1) when finetuning all models except DINO since it has a different architecture. For DINO, we start with a smaller learning rate of 0.0001 and use a smaller batch size of 16 instead.

Table B5: Finetuning on the PatchCam dataset with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50	0.5181 ± 0.0402	0.6359 ± 0.0863	0.6237 ± 0.1207	0.6631 ± 0.0578	0.7901 ± 0.0093
100	0.5725 ± 0.0692	0.6844 ± 0.1123	0.6164 ± 0.0981	0.6116 ± 0.0630	0.8115 ± 0.0300
200	0.7016 ± 0.0483	0.7656 ± 0.0395	0.6539 ± 0.0993	0.6305 ± 0.0845	0.8028 ± 0.0332
500 (S)	0.7355 ± 0.0282	0.7897 ± 0.0162	0.7895 ± 0.0336	0.8021 ± 0.0138	0.8366 ± 0.0092
1000	0.7642 ± 0.0220	0.7870 ± 0.0433	0.8174 ± 0.0182	0.8160 ± 0.0096	0.8454 ± 0.0091
2000	0.7674 ± 0.0118	0.7950 ± 0.0204	0.8221 ± 0.0255	0.7944 ± 0.0161	0.8438 ± 0.0220
5000 (M)	0.7660 ± 0.0223	0.8274 ± 0.0051	0.8399 ± 0.0142	0.8329 ± 0.0085	0.8440 ± 0.0172
10000	0.7846 ± 0.0126	0.8338 ± 0.0079	0.8338 ± 0.0163	0.8402 ± 0.0118	0.8379 ± 0.0165
20000	0.8114 ± 0.0110	0.8491 ± 0.0065	0.8587 ± 0.0116	0.8492 ± 0.0186	0.8745 ± 0.0045
Full	0.8515 ± 0.0023	0.8483 ± 0.0097	0.8619 ± 0.0090	0.8553 ± 0.0110	0.8517 ± 0.0158

Table B6: Finetuning on the APTOS dataset with various initializations. The mean Kappa score is obtained across 5 folds.

Mean Kappa Score					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50 (S)	0.0177 ± 0.0954	0.4817 ± 0.0991	0.4928 ± 0.0378	0.5916 ± 0.0570	0.6601 ± 0.0447
100	0.8080 ± 0.0759	0.5289 ± 0.0930	0.6015 ± 0.0784	0.6085 ± 0.0412	0.7144 ± 0.0709
200	0.1914 ± 0.0445	0.6634 ± 0.0405	0.7354 ± 0.0120	0.6860 ± 0.0412	0.7754 ± 0.0194
500 (M)	0.3233 ± 0.0822	0.7369 ± 0.0310	0.7594 ± 0.0246	0.7603 ± 0.0249	0.7945 ± 0.0079
Full	0.5927 ± 0.0545	0.8057 ± 0.0149	0.8293 ± 0.0133	0.8264 ± 0.0103	0.8365 ± 0.0213

Table B7: Finetuning on the Pneumonia Chest X-ray dataset with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
50 (S)	0.6895 ± 0.0512	0.8649 ± 0.0442	0.9289 ± 0.0291	0.9197 ± 0.0168	0.9256 ± 0.0235
100	0.8210 ± 0.0683	0.9032 ± 0.0423	0.9248 ± 0.0361	0.9199 ± 0.0338	0.9363 ± 0.0188
200	0.9004 ± 0.0078	0.9157 ± 0.0167	0.9593 ± 0.0125	0.9436 ± 0.0194	0.9687 ± 0.0086
500 (M)	0.9183 ± 0.0186	0.9698 ± 0.0066	0.9814 ± 0.0087	0.9781 ± 0.0085	0.9867 ± 0.0051
Full	0.9820 ± 0.0043	0.9910 ± 0.0015	0.9927 ± 0.0016	0.9950 ± 0.0013	0.9948 ± 0.0010

Table B8: Finetuning on the NIH Chest X-ray with various initializations. The mean AUC is obtained across 5 folds.

Mean AUC					
Number of samples	Random	Supervised ImageNet	SwAV	SimCLR	DINO
20 (S)	0.5015 ± 0.0253	0.5251 ± 0.0238	0.5903 ± 0.0384	0.5570 ± 0.0450	0.5552 ± 0.0546
50	0.5492 ± 0.0828	0.6105 ± 0.0381	0.6172 ± 0.0167	0.6227 ± 0.0309	0.6348 ± 0.0286
100	0.5961 ± 0.0602	0.6567 ± 0.0357	0.6616 ± 0.0436	0.6768 ± 0.0773	0.6689 ± 0.0240
150	0.6114 ± 0.0293	0.6639 ± 0.0347	0.7037 ± 0.0558	0.6795 ± 0.0515	0.6551 ± 0.0318
200 (M)	0.6404 ± 0.0165	0.6816 ± 0.0429	0.6973 ± 0.0227	0.7228 ± 0.0287	0.6652 ± 0.0114
Full	0.6616 ± 0.0345	0.7618 ± 0.0116	0.7737 ± 0.0212	0.7358 ± 0.0295	0.7404 ± 0.0240

Appendix C. Detailed results of DVME

Embedding from each self-supervised pretrained model is projected into a dimension of 512. Embeddings from SimCLR, SwAV, and DINO add up to a dimension of 1536. The self-attention module is implemented based on the timm libarary[6]. The output from the self-attention layer is further projected to a 512-dimensional layer followed by a ReLU layer, and the final linear layer. Table C1-C4 show the detailed result of linear evaluation using DVME with and without self-attention.

Table C1: Linear evaluation using DVME on the PatchCam dataset. The mean AUC is obtained across 5 folds.

Mean AUC		
Number of samples	DVME w/o self-attention	DVME
50	0.7376 ± 0.0350	0.7456 ± 0.0467
100	0.7906 ± 0.0226	0.7864 ± 0.0405
200	0.8076 ± 0.0182	0.8026 ± 0.0209
500 (S)	0.8196 ± 0.0100	0.8227 ± 0.0148
1000	0.8200 ± 0.0045	0.8316 ± 0.0112
2000	0.8242 ± 0.0083	0.8243 ± 0.0184
5000 (M)	0.8442 ± 0.0074	0.8399 ± 0.0059
10000	0.8417 ± 0.0044	0.8404 ± 0.0068
20000	0.8525 ± 0.0049	0.8444 ± 0.0100
Full	0.8478 ± 0.0052	0.8467 ± 0.0094

Table C2: Linear evaluation using DVME on the APTOS dataset. The mean Kappa score is obtained across 5 folds.

Mean Kappa Score		
Number of samples	DVME w/o self-attention	DVME
50 (S)	0.6354 ± 0.0428	0.6913 ± 0.0575
100	0.7018 ± 0.0175	0.6992 ± 0.0860
200	0.7351 ± 0.0240	0.7787 ± 0.0191
500 (M)	0.7681 ± 0.0166	0.7925 ± 0.0265
Full	0.7759 ± 0.0134	0.8242 ± 0.0279

6. https://github.com/rwightman/pytorch-image-models/blob/master/timm/models/vision_transformer.py

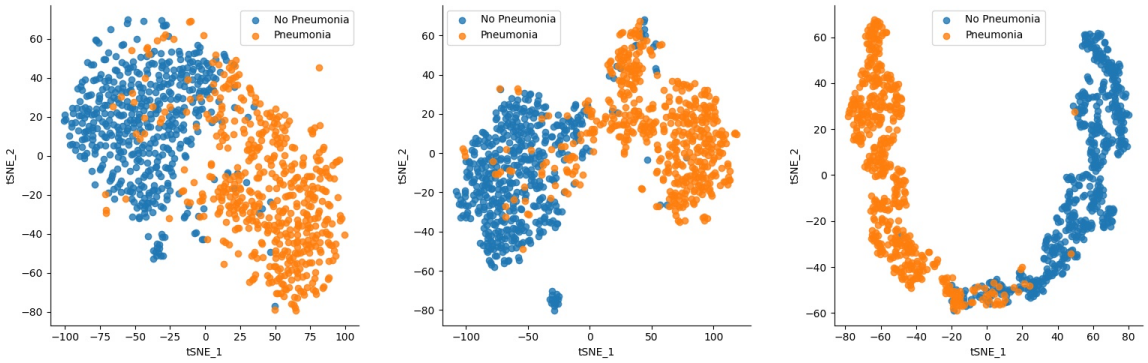
Table C3: Linear evaluation using DVME on the Pneumonia Chest X-ray dataset. The mean AUC is obtained across 5 folds.

Mean AUC		
Number of samples	DVME w/o self-attention	DVME
50 (S)	0.9543 ± 0.0072	0.9539 ± 0.0025
100	0.9528 ± 0.0128	0.9469 ± 0.0111
200	0.9532 ± 0.0177	0.9569 ± 0.0170
500 (M)	0.9725 ± 0.0030	0.9696 ± 0.0101
Full	0.9865 ± 0.0030	0.9842 ± 0.0029

Table C4: Linear evaluation using DVME on the NIH Chest X-ray dataset. The mean AUC is obtained across 5 folds.

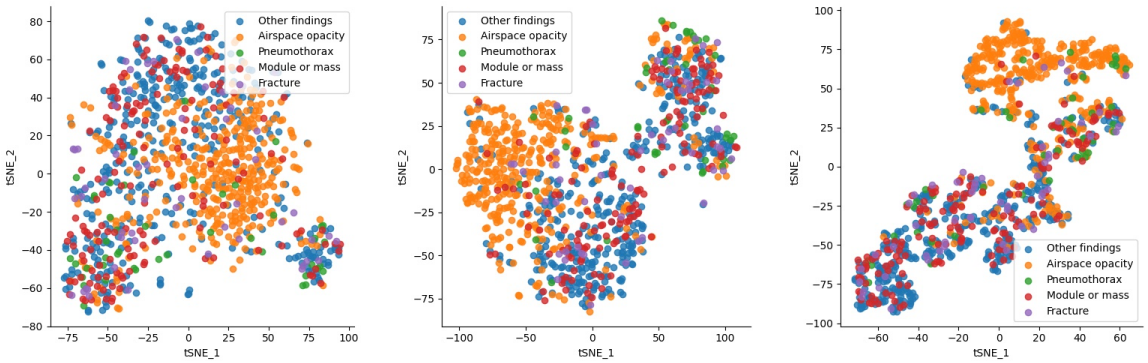
Mean AUC		
Number of samples	DVME w/o self-attention	DVME
20 (S)	0.6525 ± 0.0558	0.6566 ± 0.0564
50	0.7051 ± 0.0255	0.6871 ± 0.0400
100	0.7260 ± 0.0130	0.7091 ± 0.0428
150	0.7209 ± 0.0179	0.7437 ± 0.0310
200 (M)	0.7232 ± 0.0267	0.7601 ± 0.0146
Full	0.7575 ± 0.0177	0.7538 ± 0.0234

Appendix D. Embedding visualization



(*a*) Supervised ImageNet (*b*) DINO (*c*) DVME

Figure D1: t-SNE visualization of the pretrained embeddings from supervised ImageNet, DINO, and our proposed method DVME on **Pneumonary Chest X-ray dataset**



(*a*) Supervised ImageNet (*b*) DINO (*c*) DVME

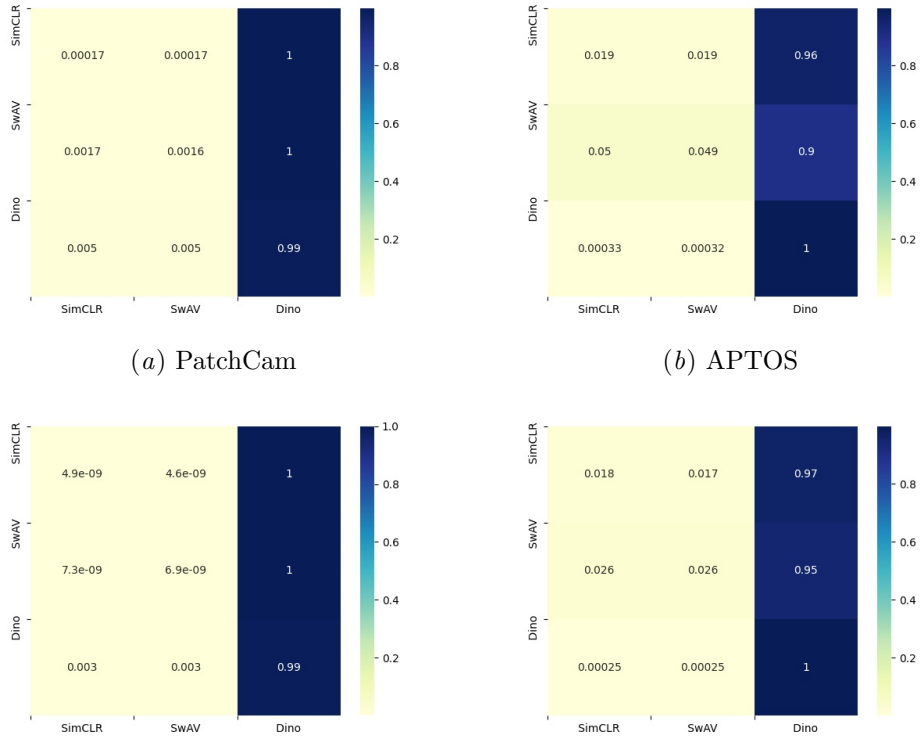
Figure D2: t-SNE visualization of the pretrained embeddings from supervised ImageNet, DINO, and our proposed method DVME on **NIH Chest X-ray dataset**

Appendix E. Dynamic Visual Meta-embeddings

```python
import torch.nn as nn
import torch

class DVME(nn.Module):

    def __init__(self, proj_dim, num_cls, attn):
        # proj_dim: dimension of projection (default is 512)
        # num_cls:  number of classes
        # attn:     self-attention module

        super(DVME, self).__init__()
        self.simclr_head = nn.Linear(2048, proj_dim)
        self.swav_head = nn.Linear(2048, proj_dim)
        self.dino_head = nn.Linear(1536, proj_dim)
        self.attn = attn
        self.normlayer = nn.LayerNorm(proj_dim*3)
        self.proj_head = nn.Linear(proj_dim*3, proj_dim)
        self.classifier = nn.Linear(proj_dim, num_cls)
        self.dropout = nn.Dropout(0.2)



    def forward(self, x):
        # x:    dictionary containing extracted embeddings from
        #       pretrained models SimCLR, SwAV, DINO

        simclr_out = self.simclr_head(x['simclr'])
        swav_out = self.swav_head(x['swav'])
        dino_out = self.dino_head(x['dino'])
        meta_x = torch.cat([simclr_out, swav_out, dino_out], dim=1)
        # reshape the meta-emb into (batch, tokens, dim)
        meta_x = meta_x.view(meta_x.size(0), -1, 1)
        out = self.attn(meta_x)
        out = self.normlayer(out.view(out.size(0), -1))
        out = self.proj_head(out).relu()
        out = self.dropout(out)
        out = self.classifier(out)
        return out
```

Listing 1: PyTorch code snippet of DVME

Appendix F. Attention weights



(*a*) PatchCam (*b*) APTOS

(*c*) Pneumonia Chest X-ray (*d*) NIH Chest X-ray

Figure F1: The self-attention weights obtained from the self-attention layers of DVME. At each dataset, the self-attention weights are averaged across all samples in test set using the models trained at the full size of dataset.

Appendix G. Number of trainable parameters

Table G1: The number of trainable parameters across all architectures in linear evaluation and finetuning.

Architecture	Evaluation Setting	Number of trainable parameters
ResNet50	Linear Evaluation	10.2 K
	Finetuning	23.5 M
ViT	Linear Evaluation	7.7 K
	Finetuning	21.7 M
DVME	Linear Evaluation	3.6 M
	Finetuning	72.4 M