Learning Low-Rank Feature for Thorax Disease Classification

Yancheng Wang[1] **Rajeev Goel**[1] **Utkarsh Nath**[1] **Alvin C. Silva**[2] **Teresa Wu**[1]
Yingzhen Yang[1]
[1] School of Computing and Augmented Intelligence, Arizona State University
{ywan1053, rgoel15, unath, teresa.wu, yingzhen.yang}@asu.edu
[2] Mayo Clinic Arizona
silva.alvin@mayo.edu

Abstract

Deep neural networks, including Convolutional Neural Networks (CNNs) and Visual Transformers (ViT), have achieved stunning success in the medical image domain. We study thorax disease classification in this paper. Effective extraction of features for the disease areas is crucial for disease classification on radiographic images. While various neural architectures and training techniques, such as self-supervised learning with contrastive/restorative learning, have been employed for disease classification on radiographic images, there are no principled methods that can effectively reduce the adverse effect of noise and background or non-disease areas on the radiographic images for disease classification. To address this challenge, we propose a novel Low-Rank Feature Learning (LRFL) method in this paper, which is universally applicable to the training of all neural networks. The LRFL method is both empirically motivated by a Low Frequency Property (LFP) and theoretically motivated by our sharp generalization bound for neural networks with low-rank features. LFP not only widely exists in deep neural networks for generic machine learning but also exists in all the thorax medical datasets studied in this paper. In the empirical study, using a neural network such as a ViT or a CNN pre-trained on unlabeled chest X-rays by Masked Autoencoders (MAE), our novel LRFL method is applied on the pre-trained neural network and demonstrates better classification results in terms of both multi-class area under the receiver operating curve (mAUC) and classification accuracy than the current state-of-the-art. The code of LRFL is available at `https://github.com/Statistical-Deep-Learning/LRFL`.

1 Introduction

Following the huge success of deep learning, recent studies have developed deep neural networks (DNNs) for various tasks in medical imaging, such as disease classification and abnormalities detection in anatomy in chest X-rays [1, 2]. Accurate clinical decision-making with DNNs heavily relies on learning informative medical feature representation. Early works adopt convolutional neural networks (CNNs) such as U-Net [3] for representation learning on radiography images. Recently, Visual Transformers (ViTs) [4] are also adopted to learn informative medical representations from radiography images [2], utilizing their capabilities in capturing long-range feature dependencies. Albeit the success of CNNs and ViTs in analyzing radiography images, their accuracy heavily relies on the quality and quantity of data and annotations [5]. However, the collection of large amounts of training data and high-quality annotations in the medical imaging domain are extremely hard [2]. To

[*] Indicates equal contribution.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-3719

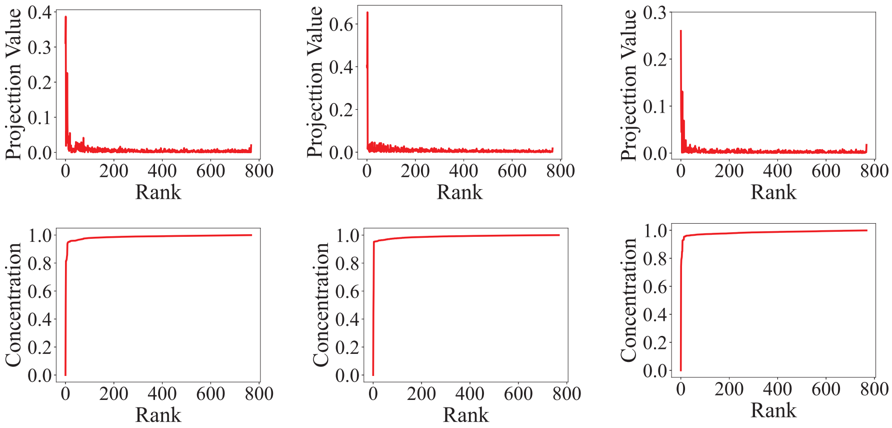
tackle this problem, self-supervised learning (SSL) has been employed as a solution for acquiring representations from unlabeled data. Given the greater availability of unlabeled medical images [6], SSL proves to be an efficient approach for obtaining discriminative representations. SSL employs a range of pretext tasks to acquire transferable representations without manual annotations. Over recent years, numerous variations of self-supervised learning have surfaced using contrastive learning [7] and restorative learning [2].

Challenges in the Current Literature for Disease Classification. We study thorax disease classification in this paper. Clinical studies show that the disease areas on radiographic images are subtle and exhibit localized variations. Such conditions are further complicated by the inevitable noise that is ubiquitous in radiographic images, as detailed in Section 2.1. Effective and robust extraction of features for the disease areas is crucial for disease classification on radiographic images. Although various neural architectures, such as CNNs and ViTs, and different training techniques, such as self-supervised learning with contrastive/restorative learning, have been employed for disease classification on radiographic images, there have been no principled methods that can effectively reduce the adverse effect of noise and background, or non-disease areas, for disease classification on radiographic images.

Our Contributions. The contributions of this paper are presented as follows. First, in order to address the aforementioned challenge, we propose a novel Low-Rank Feature Learning (LRFL) method in this paper, which is universally applicable to the training of all neural networks with the application for thorax disease classification. Our LRFL method employs low-rank features for disease classification. The usage of low-rank features is empirically motivated by a Low Frequency Property (LFP) illustrated in Figure 1. That is, the low-rank projection of the ground truth training class labels possesses the majority of the information of the training class labels. In fact, LFP widely holds for a broad range of classification problems using deep neural networks, such as [1, 8, 9]. Inspired by LFP, our LRFL method adds the truncated nuclear norm as a low-rank regularization term to the training loss of a neural network so as to perform classification using low-rank features. Because the actual features used for classification are approximately low-rank and the high-rank features are significantly truncated, all the noise and the information about the background or the non-disease areas on radiographic images in the high-rank features are largely discarded and not learned in a neural network. *Importantly and significantly different from existing low-rank learning methods reviewed in Section 2.3, we introduce a novel separable approximation for the TNN, enabling the optimization of the LRFL training loss using standard SGD*. The appropriate feature ranks retained in the LRFL method across various datasets are determined through an efficient cross-validation process, and the optimal ranks are detailed in Table 8. Extensive experimental results demonstrate that our LRFL method renders new record mAUC on three standard thorax disease datasets, NIH-ChestX-ray [10], COVIDx [11], and CheXpert [12], surpassing the current state-of-the-art [2] with the same pre-training setup.

Second, we provide a theoretical analysis showing a sharp generalization bound for the LRFL method, underscoring the substantial benefits of employing low-rank regularization within LRFL. Given these theoretical insights and the versatility of LRFL across various neural networks, we anticipate broader applications of LRFL in the classification of other diseases beyond thoracic ones, potentially enhancing classification tasks across different radiographic imaging contexts. It is worthwhile to mention that the literature has studied low-rank learning using TNN resembling LRFL, as to be reviewed in Section 2.3. Our LRFL method builds upon these foundational principles by incorporating low-rank regularization into the training of neural networks, aiming to improve thorax disease classification by reducing the adverse effects of noise and irrelevant background information. **Different from the conventional low-rank learning methods, our approach introduces a separable approximation to the TNN, facilitating the optimization process and enhancing the generalization ability of the model**. Such improved generalization is evidenced by the improved prediction accuracy of LRFL compared to the current state-of-the-art (SOTA) methods in medical image analysis.

Moreover, we have employed a conditional diffusion model trained on COVIDx and CheXpert datasets to generate synthetic images. These synthetic images are then added to their respective training sets to form the augmented training data on which our LRFL models are trained. This approach has further elevated the state-of-the-art mAUC scores achieved by LRFL on both COVIDx and CheXpert datasets.



(a) NiH-ChestXray-14

(b) COVIDx

(c) CheXpert

Figure 1: Eigen-projection (first row) and signal concentration ratio (second row) of Vit-Base on NiH-ChestXray-14, COVIDx, and CheXpert. To compute the eigen-projection, we first calculate the eigenvectors $\mathbf{U}$ of the kernel gram matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$ computed by a feature matrix $\mathbf{F} \in \mathbb{R}^{n \times d}$, then the projection value is computed by $\mathbf{p} = \frac{1}{C}\sum_{c=1}^{C}\left\|\mathbf{U}^{\top}\mathbf{Y}^{(c)}\right\|_2^2/\left\|\mathbf{Y}^{(c)}\right\|_2^2 \in \mathbb{R}^n$, where C is the number of classes, and $\mathbf{Y} \in \{0,1\}^{n \times C}$ is the one-hot labels of all the training data, $\mathbf{Y}^{(c)}$ is the c-th column of $\mathbf{Y}$. The eigen-projection $\mathbf{p}_r$ for $r \in [\min(n,d)]$ reflects the amount of the signal projected onto the r-th eigenvector of $\mathbf{K}$, and the signal concentration ratio of a rank r reflects the proportion of signal projected onto the top r eigenvectors of $\mathbf{K}$. The signal concentration ratio for rank r is computed by $\left\|\mathbf{p}^{(1:r)}\right\|_2$, where $\mathbf{p}^{(1:r)}$ contains the first r elements of $\mathbf{p}$. For example, by the rank $r = 38$, the signal concentration ratio of $\mathbf{Y}$ on NIH ChestX-ray14, COVIDx, and CheXpert are 0.959, 0.964, and 0.962 respectively.

Motivation for using synthetic images to boost the accuracy for thorax disease classification. The computer vision literature [13, 14, 15] has extensively studied the usage of the generated synthetic images which augment the training data and improve the prediction accuracy of image classification. Inspired and motivated by this observation, we propose to generate synthetic images and use them to form the augmented training data and improve the performance of thorax disease classification. The augmented training data comprise the original training images and the synthetic images. However, too many synthetic images tend to introduce more noise to the augmented training data so excessive synthetic images can hurt the prediction accuracy of DNNs trained on the augmented training data [16]. As evidenced in the ablation study in Section C.3, our proposed LRFL method, coupled with the selection of the number of synthetic images, effectively mitigates this issue. The proposed low-rank learning method only learns the low-rank part of the features learned by a deep learning model so that noise in the high-rank part would not affect the learned model. Also, cross-validation is used to select a proper number of synthetic images, which will boost the prediction accuracy while not introducing too much noise to the augmented training data.

We also present ablation study results evidencing our contributions. We compare the eigenvalues of the kernels and the kernel complexity associated with the LRFL models and the corresponding base models in Section B.4.1 of the appendix, and the lower kernel complexity of the LRFL models suggests their lower generalization error [17, 18, 19].

Notations. We use bold letters to denote matrices or vectors. $[\mathbf{A}]_i$ stands for the i-th row of a matrix $\mathbf{A}$. $\|\cdot\|_p$ denotes the p-norm of a vector or a matrix. $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm of a matrix. We use $[m \ldots n]$ to indicate numbers between m and n inclusively, and $[n]$ denotes the natural numbers between 1 and n inclusively.

2 Related Works

2.1 Radiographic Imaging

Radiographic imaging [20] is a cornerstone in medical image analysis. Unlike photographic images [21], radiography images have consistent backgrounds due to fixed imaging protocols [22, 23, 24, 2]. Clinical details are spread across the images, while areas indicating illness show localized variations [2, 25, 26], making analysis challenging. Noise is unavoidable in radiography images, stemming from quantum fluctuations, electronic interference, scatter radiation, motion blur, and overlapping structures [27, 28, 29, 30]. Quantum noise, originating from statistical fluctuations in detected X-ray photons [31, 32, 25, 30], is often the primary source. Quantum noise introduces graininess, obscuring details and diminishing contrast [31]. Modeled as a Poisson process [25, 30], it can be approximated by a Gaussian distribution under high photon flux [33, 34], enabling noise reduction techniques [34].

2.2 Medical Image Analysis with Deep Learning

Deep learning has made remarkable progress in photographic image analysis [35, 36, 37], sparking interest in applying it to medical imaging due to the ability to learn complex representations. Convolutional neural networks (CNNs) like U-Net [3, 38, 39] pioneered this field, achieving state-of-the-art performance across various tasks such as image classification [40, 41, 42], object detection [43, 39, 44], and semantic segmentation [44, 45, 39, 46, 47]. More recently, visual transformers, inspired by the success of transformers in natural language processing [48], have outperformed state-of-the-art CNNs on various computer vision benchmarks [49, 50, 4, 51, 52, 53]. Despite debates around transformers vs CNNs in terms of generalization [54, 55, 56, 57, 58], data requirements [4, 59, 60], and computational costs [61], transformers have shown great potential in medical image analysis [2, 62, 63]. Given the scarcity of high-quality annotations, self-supervised contrastive learning techniques [7, 64, 65, 66, 2] have gained traction for pre-training networks in this domain [22, 2, 62]. However, the high similarity between radiographic images due to standardized protocols [67, 68] poses challenges compared to photographic images [69, 7]. To address this, recent works utilize restorative strategies like masked autoencoders (MAE) [70, 71, 72, 73, 74, 2, 75] for pre-training [2]. Similarly, we adopt MAE [2] to pre-train our networks before learning low-rank features.

2.3 Low-Rank Learning

Low-rank learning has garnered significant attention across various fields for its capacity to reduce dimensionality, suppress noise, and enhance feature extraction. Robust Principal Component Analysis (RPCA) [76] serves as a cornerstone in this realm, efficiently separating data matrices into low-rank and sparse components. This technique proves invaluable for vision-related tasks such as image denoising and background subtraction. Building on this foundation, [77] introduced singular value pruning, a method to impose low-rank constraints on neural network layers, thereby boosting both computational efficiency and performance. The concept of TNN regularization (TNNR) has been further refined by researchers like [78], who noted that TNNR more accurately approximates the rank function by selectively minimizing singular values, essential for precise low-rank matrix recovery in noisy conditions. Following that, some existing works [79, 80, 81] propose to perform low-rank feature learning by minimizing the TNN of the feature matrix. Additionally, the use of TNNR in tensor completion has markedly improved the restoration of incomplete visual data, utilizing tensor singular value decomposition (t-SVD) [82, 83]. More contemporary learning-based methods, such as those developed by [84], have optimized low-rank approximations through targeted training, enhancing practical application outcomes. Some works [85, 86, 87] also demonstrate that learning low-rank features can significantly enhance the robustness of deep neural networks against noise in input images. In addition, recent works [88, 89, 90] find that the good generalization capabilities of deep neural networks are attributed to the fact that deeper networks are inductively biased to find solutions with lower effective rank embeddings.

3 Formulation

3.1 Pipeline for Thorax Disease Classification

We utilize the masked MAE technique [75] for the initial pre-training of both CNNs and ViTs following[2], and subsequently fine-tune the pre-trained networks with our Low-Rank Feature Learning (LRFL). The full training pipeline of learning low-rank features for disease classification can be described in three steps. In the first step, which is the **pre-training** step, we pre-train the networks using the self-supervised restorative learning method, masked MAE [75], on a diverse pre-training dataset that includes ImageNet-1k [91] and a collection of X-rays (0.5M) [2]. During this phase, we randomly mask patches on input images and drive the networks to optimize pixel-wise image reconstruction for the obscured patches. In the second step, which is the **regular fine-tuning** step, we fine-tune the pre-trained networks employing cross-entropy loss aimed at image classification on specific target datasets, namely NIH-ChestX-ray [10], COVIDx [11], and CheXpert [12]. In the last step, which is the **low-rank feature learning** step, we fix the backbones of the networks and fine-tune the linear classifier utilizing our novel LRFL method.

3.2 Problem Setup for LRFL

We now introduce the problem setup for LRFL with training details. Suppose the training data are given as $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^n$ where $\mathbf{x}_i$ and $\mathbf{y}_i \in \mathbb{R}^C$ are the i-th training data point and its corresponding class label vector respectively, and C is the number of classes. Each element $\mathbf{y}_i$ is binary with $\mathbf{y}_i = 1$ indicating the i-th disease is present in $\mathbf{x}_i$, otherwise $\mathbf{y}_i = 0$. Suppose that the neural network trained by step two of our pipeline in Section 3.1 generates a feature vector $f_{\mathbf{W}_1(0)}(\mathbf{x}) \in \mathbb{R}^d$ (the output of the layer preceding the final linear/softmax layer of the network) for any input x, and $f_{\mathbf{W}'}(\cdot)$ is the feature extraction function with $\mathbf{W}'$ being the weights of the feature extraction backbone of the network. $\mathbf{W}_1(0)$ denotes the denotes the weights of feature extraction backbone by step two of the pipeline. We can train a neural network by optimizing

$$\min_{\mathbf{W}} L(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{KL}\left(\mathbf{y}_i, \sigma\left(\mathbf{W}_2 f_{\mathbf{W}_1(0)}(\mathbf{x})\right)\right), \tag{1}$$

where $\mathbf{W}_1$ is initialized by $\mathbf{W}_1(0)$, $\mathbf{W}_2 \in \mathbb{R}^{C\times d}$, and $\mathbf{W} = (\mathbf{W}_1, \mathbf{W}_2)$. Here σ is an element-wise sigmoid function, $\sigma(\boldsymbol{a}) \in \mathbb{R}^C$ with $[\sigma(\boldsymbol{a})]_c = 1/(1+\exp(-\boldsymbol{a}_c))$ for $\boldsymbol{a} \in \mathbb{R}^C$ and $c \in [C]$. KL stands for the element-wise binary cross-entropy function. Given two nonnegative vectors $\mathbf{u} = [u_1, \ldots, u_d] \in \mathbb{R}^d, \mathbf{v} = [v_1, \ldots, v_d] \in \mathbb{R}^d$ where $u_i \in \{0,1\}$ for all $i \in [d]$ and $\|\mathbf{v}\|_\infty \le 1$, $\mathrm{KL}(\mathbf{u}, \mathbf{v}) \coloneqq \sum_{j=1}^d -u_i \log v_i - (1-u_i)\log(1-v_i)$. We use $\mathbf{Y} = \left[\mathbf{y}_1^\top; \mathbf{y}_2^\top; \ldots; \mathbf{y}_n^\top\right] \in \mathbb{R}^{n\times C}$ to denote the training label matrix by stacking the label vectors of all the training data. Let the mapping function of the neural network used in the loss function $L(\mathbf{W})$ be $\mathrm{NN}_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}_2 f_{\mathbf{W}_1}(\mathbf{x})$.

Motivation for Low-Rank Regularization The Low Frequency Property is illustrated in Figure 1, that is, the low-rank projection of the ground truth class labels possesses the majority of the information of the class labels. Inspired by this observation, our LRFL encourages the low-rank part of the feature to participate in the classification process. In this way, the noise and non-disease areas in the high-rank part of the feature are mostly not learned by LRFL so as to improve the classification accuracy. Using notations in Section 3.2, the truncated nuclear norm of $\mathbf{F}$ is $\|\mathbf{F}\|_T \coloneqq \sum_{i=T+1}^d \sigma_i$ where $T \in [0, d]$. It can be observed by the generalization error bound discussed in Section 3.2 that a smaller $\|\mathbf{F}\|_T$ renders a tighter upper bound for the generalization error of the linear neural network used for LRFL. This observation gives a strong theoretical motivation for us to add the truncated nuclear norm $\|\mathbf{F}\|_T$ to the training loss $L(\mathbf{W})$.

3.3 Generalization Bound for Low-Rank Feature Learning

We define the loss function $\ell(\mathrm{NN}_{\mathbf{W}}(\mathbf{x}), \mathbf{y}) \coloneqq \|\mathrm{NN}_{\mathbf{W}}(\mathbf{x}) - \mathbf{y}\|_2^2$, and the generalization error of the network NN is the expected risk of the loss ℓ, which is denoted by $L_{\mathcal{D}}(\mathrm{NN}_{\mathbf{W}}) \coloneqq \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\left[\ell(\mathrm{NN}_{\mathbf{W}}(\mathbf{x}), \mathbf{y})\right]$, with $\mathcal{D}$ being the distribution of the data $\mathbf{x}$ and its class label $\mathbf{y}$. The network $\mathrm{NN}_{\mathbf{W}}$ generates a feature $\mathbf{F} \in \mathbb{R}^{n\times d}$ of all the training data with $\mathbf{F}_i = f_{\mathbf{W}_1}^\top(\mathbf{x}_i)$ for $i \in [n]$. The kernel gram matrix for the feature $\mathbf{F}$ is $\mathbf{K}_n = \frac{1}{n}\mathbf{F}\mathbf{F}^\top$. We let $\widehat{\lambda}_1 \ge \widehat{\lambda}_2 \ge \ldots \ge \widehat{\lambda}_{\bar{r}} > 0$ where $\bar{r} \le \min\{n, d\}$ is the rank of $\mathbf{K}_n$. Suppose the Singular Value Decomposition of $\mathbf{F}$ is $\mathbf{F} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{n\times d}$ has orthogonal columns, $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ is a diagonal matrix with diagonal elements being the singular values of $\mathbf{F}$, and $\mathbf{V} \in \mathbb{R}^{d\times d}$ is an orthogonal matrix. The columns of $\mathbf{U}$ and $\mathbf{V}$ are also called the left eigenvectors and the right eigenvectors of $\mathbf{F}$, respectively. Let $\sigma_1 \ge \sigma_2 \ldots \ge \sigma_d$ be the singular values of $\mathbf{F}$, and $\bar{\mathbf{Y}} = \mathbf{U}^{(\bar{r})}{\mathbf{U}^{(\bar{r})}}^\top\mathbf{Y}$ be the projection of the training label matrix $\mathbf{Y}$ onto the subspace spanned by the top-$\bar{r}$ left eigenvectors of $\mathbf{F}$, where $\mathbf{U}^{(\bar{r})} \in \mathbb{R}^{n\times\bar{r}}$ is formed by the top $\bar{r}$ eigenvectors in $\mathbf{U}$. Then, we have the following theorem giving the sharp generalization error bound for the linear neural network in (1).

Theorem 3.1. For every $x > 0$, with probability at least $1 - \exp(-x)$, after the t-th iteration of gradient descent for all $t \ge 1$, we have

$$L_{\mathcal{D}}(\mathrm{NN}_{\mathbf{W}}) \le \left\|\mathbf{Y} - \bar{\mathbf{Y}}\right\|_{\mathrm{F}} + c_1\left(1 - \eta\widehat{\lambda}_r\right)^{2t}\|\mathbf{Y}\|_{\mathrm{F}}^2 \quad + c_2 \min_{h\in[0,r]}\left(\frac{h}{n} + \sqrt{\frac{1}{n}\sum_{i=h+1}^{r}\widehat{\lambda}_i}\right) + \frac{c_3 x}{n}, \tag{2}$$

where c_1, c_2, c_3 are positive constants.

Remark 3.2. The RHS of (2) is the generalization error bound for the linear neural network used in LRFL as step three of the pipeline in Section 3.1. Moreover, let $\sigma_1 \ge \sigma_2 \ldots \ge \sigma_d$ be the singular values of $\mathbf{F}$. Due to the fact that $\sqrt{\frac{1}{n} \sum\limits_{i=h+1}^{r} \widehat{\lambda}_i} \le \frac{1}{n} \sum\limits_{i=h+1}^{r} \sigma_i$, it follows by (2) that

$$L_{\mathcal{D}}(\mathrm{NN}_{\mathbf{W}}) \le c_1 \left(1 - \eta \widehat{\lambda}_r\right)^{2t} \|\mathbf{Y}\|_{\mathrm{F}}^2 + c_2 \left(\frac{h}{n} + \frac{1}{n} \sum_{i=T+1}^{d} \sigma_i\right) + \frac{c_3 x}{n}, \tag{3}$$

which holds for all $T \in [0, d]$. (3) motivates the reduction of the truncated nuclear norm of the feature $\mathbf{F}$, as detailed in the next subsection.

3.4 Optimization of the Truncated Nuclear Norm in SGD

The truncated nuclear norm $\|\mathbf{F}\|_T$ is not separable, so the training loss with $\|\mathbf{F}\|_T$ cannot be directly optimized by the standard SGD. To address this problem, we propose an approximation $\overline{\|\mathbf{F}\|_T}$ to $\|\mathbf{F}\|_T$ which is separable so that $\overline{\|\mathbf{F}\|_T}$ can be optimized by standard SGD.

First, we note that if $\mathbf{U}, \mathbf{V}$ are known, then $\mathbf{\Sigma} = \mathbf{U}^\top \mathbf{F} \mathbf{V}$. If we have an approximation $\overline{\mathbf{U}}$ to $\mathbf{U}$ and an approximation $\overline{\mathbf{V}}$ to $\mathbf{V}$, then $\mathbf{\Sigma}$ can be approximated by $\overline{\mathbf{\Sigma}} = \overline{\mathbf{U}}^\top \mathbf{F} \overline{\mathbf{V}}$. As a result, the approximation $\overline{\|\mathbf{F}\|_T}$ to the truncated nuclear norm is $\overline{\|\mathbf{F}\|_T} = \sum\limits_{i=1}^{n} \left(\sum\limits_{s=T+1}^{d} \sum\limits_{k=1}^{d} \overline{\mathbf{U}}_{si}^\top \mathbf{F}_{ik} \overline{\mathbf{V}}_{ks} \right)$. Due to the above discussions, the loss function of LRFL with the approximate truncated nuclear norm $\overline{\|\mathbf{F}\|_T}$ is $\mathcal{L}_{\mathrm{LRFL}}(\mathbf{W}) = \frac{1}{m} \sum_{v_i \in \mathcal{V}_{\mathcal{L}}} \mathrm{KL}\left(\mathbf{y}_i, \left[\sigma\left(\mathbf{F}\mathbf{W}^{(\mathrm{lin})}\right)\right]_i\right) + \eta \overline{\|\mathbf{F}\|_T}$, which is separable, so that it can be trained by the standard SGD. $\eta > 0$ is the weighting parameter for the truncated nuclear norm. Because $\mathcal{L}_{\mathrm{LRFL}}(\mathbf{W})$ is to be optimized by the standard SGD, we have the loss function of LRFL for the j-th minibatch $\mathcal{B}_j \subseteq [n]$ as

$$\mathcal{L}_j(\mathbf{W}) = \frac{1}{|\mathcal{B}_j|} \sum_{i \in \mathcal{B}_j} \mathrm{KL}\left(\mathbf{y}_i, \left[\sigma\left(\mathbf{F}\mathbf{W}^{(\mathrm{lin})}\right)\right]_i\right) + \frac{\eta}{|\mathcal{B}_j|} \sum_{i \in \mathcal{B}_j} \left(\sum_{s=T+1}^{d} \sum_{k=1}^{d} \overline{\mathbf{U}}_{si}^\top \mathbf{F}_{ik} \overline{\mathbf{V}}_{ks} \right). \tag{4}$$

The approximation $\overline{\mathbf{U}}$ and $\overline{\mathbf{V}}$ can be computed as the left and right eigenvectors of the feature $\mathbf{F}$ computed at earlier epochs. In order to save computation and avoiding performing SVD for $\mathbf{F}$ at every epoch, we propose to update $\overline{\mathbf{U}}$ and $\overline{\mathbf{V}}$ only after certain epochs. Algorithm 1 describes the training algorithm for the neural network trained with LRFL, which uses the standard SGD to optimize the loss function $\mathcal{L}_{\mathrm{LRFL}}(\mathbf{W})$, as step three of our pipeline in Section 3.1. Before the first epoch, we compute $\overline{\mathbf{U}}$ and $\overline{\mathbf{V}}$ as the left and right eigenvectors of the feature $\mathbf{F}$ at the initialization of the neural network. After every t_0 epoch with t_0 being a constant integer, we update $\overline{\mathbf{U}}$ and $\overline{\mathbf{V}}$ as the left and right eigenvectors of the feature $\mathbf{F}$ produced by the neural network right after t_0-th epoch, with t_0 being a constant integer.

Algorithm 1 Training Algorithm with the Approximate Truncated Nuclear Norm by SGD

```
Initialize the weights $\mathbf{W}_1$ by $\mathbf{W}_1 = \mathbf{W}_1(0)$, and initialize $\mathbf{W}_2$ randomly
Compute feature $\mathbf{F}$ by the neural network, and its SVD as $\mathbf{F} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}$
Update $\overline{\mathbf{U}} = \mathbf{U}, \overline{\mathbf{V}} = \mathbf{V}$
for $t = 1, 2, \ldots, t_{\max}$ do
    if $t \equiv 0 \pmod{t_0}$ then
        Compute feature $\mathbf{F}$ of the neural network, and its SVD $\mathbf{F} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}$.
        Update $\overline{\mathbf{U}} = \mathbf{U}, \overline{\mathbf{V}} = \mathbf{V}$
    end if
    for $b = 1, 2, \ldots, B$ do
        Update $\mathbf{W}$ by applying gradient descent on batch $\mathcal{B}_j \subseteq [n]$ using the gradient of the loss $\mathcal{L}_j$ in Eq.(4)
    end for
end for
return The trained weights $\mathbf{W}$ of the network
```

4 Experimental Results

In this section, we conduct experiments on medical datasets to demonstrate the effectiveness of the proposed LRFL method. The experiments section is organized as follows: In Section 4.1, we

https://doi.org/10.52202/079017-3719

discuss our experimental setup and implementation details. In Sections 4.2 and 4.3, we evaluate the LRFL models for thorax disease classification on CheXpert and COVIDx. Evaluation results on NIH ChestX-ray14 are deferred to Section B.1 of the appendix. In Section 4.4, we evaluate synthetic data augmentation on LRFL models, with additional details and results deferred to Section C of the appendix. Comprehensive ablation studies on LRFL are performed in Section 4.5. In Section 4.5.1, we study the effectiveness of the LRFL models in reducing the adverse effect of the background for disease classification. In Section 4.5.2, we study the performance of LRFL models for disease localization. Grad-CAM visualization results of LRFL models and baseline models are illustrated in Section 4.5.3. Additional ablation studies are deferred to Section B.4 of the appendix. In Section B.4.1, we compare the kernel eigenvalues and kernel complexity between the LRFL models and their corresponding base models to show that LRFL improves the generalization capability of the base models by reducing their kernel complexity. In Section B.4.2, we evaluate the performance of the LRFL models with limited data availability. In Section B.4.3, we compare the performance of the LRFL with other fine-tuning strategies. In Section B.4.5, we present the training time of the LRFL models compared with the corresponding base models.

4.1 Implementation Details

In this section, we evaluate the proposed LRFL for thorax disease classification. We utilize networks pre-trained on ImageNet [92] or chest X-rays in [2] with MAE, a self-supervised learning strategy that reconstructs missing pixels from patches of input images. We fine-tune these pre-trained networks with low-rank regularization on three public X-ray datasets: (1) NIH ChestX-ray14 [10], (2) Stanford CheXpert [12], and (3) COVIDx [11]. The ADAM optimizer is used with a batch size of 1024 for all datasets. Initially, we fine-tune the entire networks for 75 epochs following the settings in [2], then fine-tune with low-rank regularization for another 75 epochs. We use a cosine learning rate schedule, and the initial learning rate, which is denoted as μ, is selected by cross-validation for each model and each dataset. The default values for momentum and weight decay are set to 0.9 and 0, respectively. We use standard data augmentation techniques, including random-resize cropping, random rotation, and random horizontal flipping. For a fair comparison, all baselines are also fine-tuned for an additional 150 epochs, showing almost no improvement. An exhaustive analysis of this additional fine-tuning is in Section B.4.3. We evaluate our LRFL method on both CNN and visual transformer architectures, including ResNet-50, DenseNet, ViT-S, and ViT-B. Our model is referred to as 'X-LR', where X is the base model (e.g., ResNet-50-LR for ResNet-50 with low-rank features).

Tuning the T, η, and μ by Cross-Validation. We search for the optimal values of feature rank T, the weighting parameter for the truncated nuclear norm η, and the learning rate μ on each dataset. Let $T = \lceil \gamma \min(n, d) \rceil$, where γ is the rank ratio. We select the values of γ and η by performing 5-fold cross-validation on 20% of the training data in each dataset. The value of γ is selected from $\{0.01, 0.02, 0.03, 0.04, 0.05, 0.1, 0.15, 0.2\}$. The value of η is selected from $\left\{5 \times 10^{-4}, 1 \times 10^{-3}, 2.5 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}\right\}$. The value of μ is selected form $\left\{5 \times 10^{-4}, 2.5 \times 10^{-4}, 1 \times 10^{-4}, 5 \times 10^{-5}, 2.5 \times 10^{-5}, 1 \times 10^{-5}\right\}$. To determine the optimal values of the parameters η, γ, and μ, we employ a sequential greedy search strategy. We first fix η and μ and find the optimal value of γ by cross-validation. Subsequently, using this optimized γ, we proceed to search for the optimal η while keeping μ constant. Finally, with optimal γ and η, we search for the optimal μ by cross-validation. The optimal values of η, γ, and μ selected by cross-validation are shown in Table 8 in Section B.3 of the appendix. The time spent for the entire cross-validation process is presented in Table 9 Section B.3 of the appendix, which demonstrates that the cross-validation process is efficient and does not significantly increase the computational overhead of the training process.

4.2 Stanford CheXpert

Experimental setup. CheXpert [12] consists of 224,316 chest X-rays collected from 65,240 patients, where 191,028 chest X-rays are used for training. Each X-ray in the dataset has radiology reports indicating the presence of 14 diseases. Following the protocol in [2], all images are resized into 224×224. We also report the mean AUC (Area Under the Curve) for the 5 distinct classes and conduct a comprehensive comparison with state-of-the-art baseline methods.

Results and analysis. Table 1 presents the performance comparisons between the baseline models and the LRFL models on the CheXpert dataset. Throughout this section, we use the postfix "-LR" to

https://doi.org/10.52202/079017-3719

indicate a neural network trained with our LRFL. For example, we use the ViT-B model pre-trained on $489,090$ and the ViT-S model pre-trained on $266,340$ chest X-rays with Masked Autoencoders (MAE) [2]. The pre-trained ViT-B network is fine-tuned on the CheXpert dataset and achieves a mean AUC of 89.3. It is observed that ViT-B-LR achieves state-of-the-art performance of 89.8% in mAUC and improves the performance of ViT-B by 0.5% in mAUC. ViT-S-LR also improves the performance of ViT-S by 0.4% in mAUC, which demonstrates the power of LRFL. We also show the classification accuracy of the five diseases in Table 1, where our method exhibits much better performance than baseline methods. For example, ViT-S-LR achieves an mAUC of 86.3% on Cardiomegaly, with a 4.5% improvement over ViT-S trained with MAE. Such improvements demonstrate the power of LRFL in detecting distinct diseases. The comparison between LRFL models and a more comprehensive list of baseline models are deferred to Table 7 of the appendix.

Table 1: Performance comparisons between LRFL models and SOTA baselines on CheXpert. The best result is highlighted in bold, and the second-best result is underlined. This convention is followed by all the tables in this paper. DN represents DenseNet.

Method	Architecture	Rank	Atelectasis	Cardiomegaly	Consolidation	Edema	Effusion	mAUC (%)
Irvin et al.[12]		-	81.8	82.8	93.8	93.4	92.8	88.9
Pham et al.[9]	DN121	-	82.5	85.5	93.7	93.0	92.3	89.4
Kang et al.[93]	DN121	-	82.1	85.9	**94.4**	89.2	93.6	89.0
MoCo v2 [2]	DN121	-	78.5	77.9	92.5	92.8	92.7	88.7
ViT-S [2]	ViT-S/16	-	83.5	81.8	93.5	94.0	93.2	89.2
ViT-S-LR (Ours)	ViT-S/16	0.05r	**83.7**	86.3	90.9	93.7	93.1	89.6
ViT-B [2]	ViT-B/16	-	82.7	83.5	92.5	93.8	**94.1**	89.3
ViT-B-LR (Ours)	ViT-B/16	0.05r	81.6	85.4	93.4	**94.6**	93.9	**89.8**

4.3 COVIDx

Experimental setup. COVIDx (Version 9A) [11] consists of 30,386 chest X-rays collected from 17,026 unique patients. We follow the previous works [11, 2] in splitting the dataset into 29,986 training images with four different classes and 400 testing images with three classes. For fair comparisons with the previous methods, we report Top-1 accuracy on the test set (3 classes).

Results and Analysis. Table 2 compares the performance of SOTA transformer-based models and the LRFL models on the COVIDx dataset. Similar to Section 4.2, the base ViTs are first pre-trained on chest X-rays using Masked Autoencoders (MAE), and then the pre-trained model is fine-tuned on the COVIDx dataset. It can be observed from Table 2 that both ViT-S-LR and ViT-B-LR outperform their corresponding base models ViT-S and ViT-B, achieving an increase in accuracy of 1.6% and 1.7%, respectively. Table 2 also compares the performance of our LRFL models against the state-of-the-art models on the COVIDx dataset. LRFL models achieve much higher accuracy compared to CNN-based models such as DenseNet-121. ViT-B-LR achieves the new SOTA performance of 97% top-1 accuracy with input resolution set to 224×224, which exceeds the previous SOTA performance [2] by 1.7% in top-1 accuracy.

Table 2: Performance comparisons between LRFL models and SOTA baselines on COVIDx (in accuracy). DN represents DenseNet.

Method	Architecture	Rank	Covid-19 Sensitivity	Accuracy
COVIDNet-CXR Small [8]	-	-	87.1	92.6
COVIDNet-CXR Large [8]	-	-	96.8	94.4
MoCo v2 [2]	DN121	-	94.5	94.0
DN121 [2]	DN121	-	97.0	93.5
ViT-S [2]	ViT-S/16	-	94.5	95.2
ViT-S-LR (Ours)	ViT-S/16	0.01r	97.5	96.8
ViT-B [2]	ViT-B/16	-	95.5	95.3
ViT-B-LR (Ours)	ViT-B/16	0.003r	**98.5**	**97.0**

4.4 Improved Results using Diffusion Model

Experimental Setup. In this section, we aim to further improve the performance of LRFL models by adding labeled synthetic radiographic images of thorax diseases to the training sets of COVIDx and CheXpert. The synthetic radiographic images are generated by a conditional diffusion model, Diffusion Transformer (DiT) [94], trained on the training set of the corresponding dataset. Details on the training of DiT are deferred to Section C.2 of the appendix. To maintain the same disease co-occurrence, synthetic radiographic images are generated based on the labels from the label set of each dataset. The number of synthetic images added to the training set of each dataset is determined via cross-validation. We first generate synthetic images of the same size as the training set. The optimal percentage of synthetic images is selected using 5-fold cross-validation on the training data, which is detailed in Section C.2 of the appendix. Synthetic images are combined with the original dataset for further fine-tuning with low-rank regularization. Ablation studies on the number of synthetic images incorporated are performed in SectionC.3.

Results. The results of LRFL models trained after adding synthetic images on CheXpert and COVIDx are shown in Table 3. It is observed from the results that adding synthetic data into the training set of LRFL models can further increase their performance. For example, ViT-B-LR with synthetic images added in training outperforms the corresponding base model ViT-B by 2.2% on COVIDx.

Table 3: Performance comparison of baseline models and LRFL models on the CheXpert and COVIDx datasets, with and without synthetic data. n denotes the number of training images in the respective dataset.

Method	Architecture	CheXpert			COVIDx		
		Rank	# Synthetic Images	mAUC (%)	Rank	# Synthetic Images	Accuracy (%)
ViT-S [2]	ViT-S/16	-	-	89.2	-	-	95.2
ViT-S-LR (Ours)	ViT-S/16	0.05r	-	89.6	0.01r	-	96.8
ViT-S (Ours)	ViT-S/16	-	$0.2n$	89.3	-	$1.0n$	97.0
ViT-S-LR (Ours)	ViT-S/16	0.05r	$0.2n$	89.7	0.01r	$1.0n$	97.3
ViT-B [2]	ViT-B/16	-	-	89.3	-	-	95.3
ViT-B-LR (Ours)	ViT-B/16	0.025r	-	89.8	0.003r	-	97.0
ViT-B (Ours)	ViT-B/16	-	$0.25n$	89.9	-	$1.0n$	97.0
ViT-B-LR (Ours)	ViT-B/16	0.025r	$0.25n$	**90.4**	0.003r	$1.0n$	**97.5**

4.5 Ablation Study

4.5.1 Study of LRFL in Reducing the Adverse Effects of Background

To demonstrate that the LRFL models are more robust to the background than the baselines, we perform an ablation study on the LRFL to reduce the adverse effects of the background. In this study, we create a mask for the disease area for each original image, then decompose the original image, which has a bounding box for the disease, into a disease image and a background image. Both the disease image and the background image are of the same size as the original image. The background image has grayscale 0 in the masked disease area, and the disease image has grayscale 0 in the non-disease area. We feed the three images, which are the original image, the disease image, and the background image, to an LRFL model and obtain the original features, disease features, and background features for the LRFL model, respectively. We also feed these three images to a baseline model and obtain the original features, disease features, and background features for the baseline model. For each original image, we measure the distance between the disease features and original features using KL-divergence on the softmaxed features for the LRFL model and the baseline model. We then compute the average feature distance for each model, which is the average distance between the disease features and original features over the images with a ground-truth bounding box for the disease in the NIH ChestX-ray 14. The results in Table 4 indicate that the original features are closer to the disease features by the LRFL models compared to the baseline models, evidencing the effectiveness of the LRFL models in reducing the adverse effect of the background area. We also remark that since only the low-rank part of the original features participates in the classification process, the noise and non-disease areas in the high-rank part of the features are mostly not learned by LRFL, and in this manner, LRFL is robust to both noise and background.

Table 4: Average feature distance between original features and disease features of images with a ground-truth bounding box for the disease in the NIH ChestX-ray 14.

Methods	mAUC (%)	Average Feature Distance
ViT-S	82.3	0.7030
ViT-S-LR	**82.7**	**0.6352**
ViT-B	83.0	0.5642
ViT-B-LR	**83.4**	**0.6628**

4.5.2 Disease Localization

To study which part of the X-ray image is responsible for the model prediction by the LRFL models, we perform the disease localization experiment following the settings in [2]. We first obtain the Grad-

CAM visualization results with the last transformer block of ViT-S. The experiments are performed with all the images with a ground-truth bounding box for disease in ChestX-ray14. The predicted bounding box is generated with the thresholded Grad-CAM heatmap, largest connected component, and box regression. We evaluate the performance of disease localization by Intersection over Union (IoU) between the ground-truth bounding box and the predicated bounding box used for evaluation. The Average Precision (AP) on 25% and 50% IoUs, which are denoted as AP_{25} and AP_{50}, for ViT-S and ViT-S-LR are shown in Table 5. It is observed from the results that our LRFL model significantly outperforms the base model in detecting the bounding box of thorax disease. For example, ViT-S-LR outperforms ViT-S by 26.9% in AP_{25} for detecting the bounding box of Mass. In addition, ViT-S-LR outperforms ViT-S by 21.2% in AP_{25} for detecting the bounding box of Effusion.

Table 5: AP_{25} and AP_{50} scores for different diseases using ViT-S and ViT-S-LR models.

Disease	Size (# of px)	AP_{25}		AP_{50}	
		ViT-S	ViT-S-LR	ViT-S	ViT-S-LR
Mass	756	27.0	**53.9**	**11.1**	8.0
Atelectasis	924	31.5	**49.3**	8.1	**11.3**
Pneumothorax	1899	4.7	**18.3**	0.0	**1.5**
Infiltrate	2754	11.4	**22.7**	1.3	**2.1**
Effusion	2925	8.8	**30.0**	1.0	**3.1**
Pneumonia	2944	27.8	**44.1**	9.3	**12.5**
All	2300	18.0	**28.5**	4.7	**5.2**

4.5.3 Grad-CAM Visualization

To study how LRFL improves the performance of base models in disease detection, we use the Grad-CAM [95] to visualize the parts in the input images that are responsible for the predictions of the base models and low-rank models. Robust Grad-CAM [95] visualization results of Low-Rank ViT-Base are illustrated in Figure 2. All Grad-CAM visualization results illustrate that our LRFL models usually focus more on the areas inside the bounding box associated with the labeled disease. In contrast, the base models also focus on the areas outside the bounding box or even areas in the background. Robust Grad-CAM visualization results of Low-Rank ResNet-50 and additional Grad-CAM visualization results of Low-Rank ViT-Base are deferred to Figure 4 and Figure 5 in Section B.4.4 of the appendix.

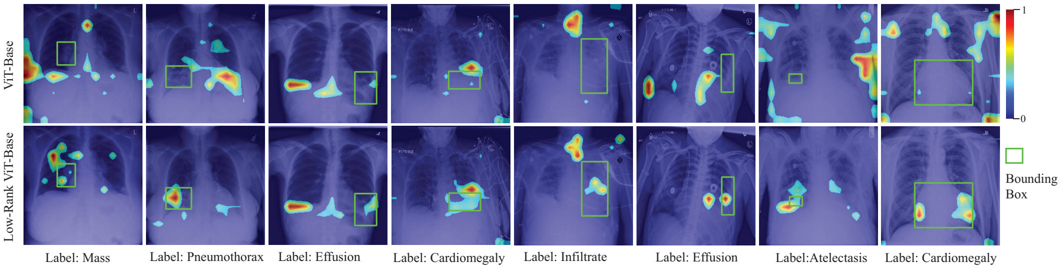


Figure 2: Robust Grad-CAM [95] visualization results on NIH ChestX-ray 14. The figures in the first row are the visualization results of ViT-Base, and the figures in the second row are the visualization results of Low-Rank ViT-Base.

5 Conclusion

In this paper, we propose a novel Low-Rank Feature Learning (LRFL) method for thorax disease classification, which can effectively reduce the adverse effect of noise and background, or non-disease areas, on the radiographic images for disease classification. Being universally applicable to the training of all neural networks, LRFL is both empirically motivated by the low frequency property and theoretically motivated by our sharp generalization bound for neural networks with low-rank features. Extensive experimental results on thorax disease datasets, including NIH-ChestX-ray, COVIDx, and CheXpert, demonstrate the superior performance of LRFL in terms of mAUC and classification accuracy. In addition, the performance of LRFL models is further improved by adding synthetic radiographic images into the training set for data augmentation.

Acknowledgments and Disclosure of Funding

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 17STQAC00001-07-00. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security. This work is also partially supported by the 2023 Mayo Clinic and Arizona State University Alliance for Health Care Collaborative Research Seed Grant Program under Grant Award Number AWD00038846.

References

[1] Sebastian Guendel, Sasa Grbic, Bogdan Georgescu, Siqi Liu, Andreas Maier, and Dorin Comaniciu. Learning to recognize abnormalities in chest x-rays with location-aware dense networks. In *Iberoamerican Congress on Pattern Recognition*, pages 757–765. Springer, 2018.

[2] Junfei Xiao, Yutong Bai, Alan Yuille, and Zongwei Zhou. Delving into masked autoencoders for multi-label thorax disease classification. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3588–3600, 2023.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020.

[5] Ruibin Feng, Zongwei Zhou, Michael B Gotway, and Jianming Liang. Parts2whole: Self-supervised contrastive learning via reconstruction. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*, 2020.

[6] Shekoofeh Azizi, Laura Culp, Jan Freyberg, Basil Mustafa, Sebastien Baur, Simon Kornblith, Ting Chen, Patricia MacWilliams, S Sara Mahdavi, Ellery Wulczyn, et al. Robust and efficient medical imaging with self-supervision. *arXiv preprint arXiv:2205.09723*, 2022.

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

[8] Linda Wang, Zhong Qiu Lin, and Alexander Wong. Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific Reports*, 10(1):19549, Nov 2020.

[9] Hieu H Pham, Tung T Le, Dat Q Tran, Dat T Ngo, and Ha Q Nguyen. Interpreting chest x-rays via cnns that exploit hierarchical disease dependencies and uncertainty labels. *Neurocomputing*, 437:186–194, 2021.

[10] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

[11] Maya Pavlova, Tia Tuinstra, Hossein Aboutalebi, Andy Zhao, Hayden Gunraj, and Alexander Wong. Covidx cxr-3: a large-scale, open-source benchmark dataset of chest x-ray images for computer-aided covid-19 diagnostics. *arXiv preprint arXiv:2206.03671*, 2022.

[12] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597, 2019.

https://doi.org/10.52202/079017-3719

[13] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *Transactions on Machine Learning Research*, 2023.

[14] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip H. S. Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023.

[15] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*, 2024.

[16] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *Trans. Mach. Learn. Res.*, 2023, 2023.

[17] Peter L. Bartlett, Olivier Bousquet, and Shahar Mendelson. Local rademacher complexities. *Ann. Statist.*, 33(4):1497–1537, 08 2005.

[18] Vladimir Koltchinskii. Local rademacher complexities and oracle inequalities in risk minimization. *Ann. Statist.*, 34(6):2593–2656, 12 2006.

[19] Shahar Mendelson. Geometric parameters of kernel machines. In *Conference on Learning Theory (COLT)*, 2002.

[20] Jun Li, Junyu Chen, Yucheng Tang, Ce Wang, Bennett A Landman, and S Kevin Zhou. Transforming medical imaging with transformers? a comparative review of key properties, current progresses, and future perspectives. *Medical image analysis*, page 102762, 2023.

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

[22] Zongwei Zhou. *Towards Annotation-Efficient Deep Learning for Computer-Aided Diagnosis*. PhD thesis, Arizona State University, 2021.

[23] Jun Li, Junyu Chen, Yucheng Tang, Bennett A Landman, and S Kevin Zhou. Transforming medical imaging with transformers? a comparative review of key properties, current progresses, and future perspectives. *arXiv preprint arXiv:2206.01136*, 2022.

[24] Fahad Shamshad, Salman Khan, Syed Waqas Zamir, Muhammad Haris Khan, Munawar Hayat, Fahad Shahbaz Khan, and Huazhu Fu. Transformers in medical imaging: A survey. *arXiv preprint arXiv:2201.09873*, 2022.

[25] Paul Suetens. Fundamentals of medical imaging. 2017.

[26] Zongwei Zhou, Michael Gotway, and Jianming Liang. Interpreting medical images. In *Intelligent Systems in Medicine and Health: The Role of AI*. 2022.

[27] JH Siewerdsen, LE Antonuk, Y El-Mohri, J Yorkston, W Huang, JM Boudry, and IA Cunningham. Empirical and theoretical investigation of the noise performance of indirect detection, active matrix flat-panel imagers (amfpis) for diagnostic radiology. *Medical physics*, 24(1):71–89, 1997.

[28] JH Siewerdsen, LE Antonuk, Y El-Mohri, J Yorkston, W Huang, and IA Cunningham. Signal, noise power spectrum, and detective quantum efficiency of indirect-detection flat-panel imagers for diagnostic radiology. *Medical physics*, 25(5):614–628, 1998.

[29] EN Manson, V Atuwo Ampoh, E Fiagbedzi, JH Amuasi, JJ Flether, and C Schandorf. Image noise in radiography and tomography: Causes, effects and reduction techniques. *Curr. Trends Clin. Med. Imaging*, 2(5):555620, 2019.

[30] Tej Bahadur Chandra and Kesari Verma. Analysis of quantum noise-reducing filters on chest x-ray images: A review. *Measurement*, 153:107426, 2020.

https://doi.org/10.52202/079017-3719

[31] K Kirk Shung, Michael Smith, and Benjamin MW Tsui. Principles of medical imaging. 2012.

[32] K Kirk Shung, Michael Smith, and Benjamin MW Tsui. Principles of medical imaging. 2012.

[33] Sangyoon Lee, Min Seok Lee, and Moon Gi Kang. Poisson–gaussian noise analysis and estimation for low-dose x-ray images in the nsct domain. *Sensors*, 18(4):1019, 2018.

[34] Qiaoqiao Ding, Yong Long, Xiaoqun Zhang, and Jeffrey A Fessler. Statistical image reconstruction using mixed poisson-gaussian noise model for x-ray ct. *arXiv preprint arXiv:1801.09533*, 2018.

[35] 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.

[36] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

[37] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[38] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, page 1, 2018.

[39] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 2018.

[40] Yan Shen and Mingchen Gao. Dynamic routing on deep neural network for thoracic disease classification and sensitive area localization. In *International Workshop on Machine Learning in Medical Imaging*, pages 389–397. Springer, 2018.

[41] Hongyu Wang, Haozhe Jia, Le Lu, and Yong Xia. Thorax-net: an attention regularized deep neural network for classification of thoracic diseases on chest radiography. *IEEE journal of biomedical and health informatics*, 24(2):475–485, 2019.

[42] Congbo Ma, Hu Wang, and Steven C. H. Hoi. Multi-label thoracic disease image classification with cross-attention networks, 2020.

[43] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.

[44] Ruixin Yang and Yingyan Yu. Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis. *Frontiers in oncology*, 11:638182, 2021.

[45] Yuan Yao, Fengze Liu, Zongwei Zhou, Yan Wang, Wei Shen, Alan Yuille, and Yongyi Lu. Unsupervised domain adaptation through shape modeling for medical image segmentation. In *Medical Imaging with Deep Learning*, 2021.

[46] Amber L Simpson, Michela Antonelli, Spyridon Bakas, Michel Bilello, Keyvan Farahani, Bram Van Ginneken, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, et al. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *arXiv preprint arXiv:1902.09063*, 2019.

[47] Jamshid Sourati, Ali Gholipour, Jennifer G Dy, Xavier Tomas-Fernandez, Sila Kurugol, and Simon K Warfield. Intelligent labeling based on fisher information for medical image segmentation using deep learning. *IEEE transactions on medical imaging*, 38(11):2642–2653, 2019.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

https://doi.org/10.52202/079017-3719

[49] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 558–567, 2021.

[50] Yancheng Wang, Ping Li, and Yingzhen Yang. Visual transformer with differentiable channel selection: An information bottleneck inspired approach, 2024.

[51] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[52] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *ICLR*, 2021.

[53] Han Cai, Chuang Gan, and Song Han. Efficientvit: Enhanced linear attention for high-resolution low-computation visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[54] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11963–11975, 2022.

[55] Yutong Bai, Jieru Mei, Alan L Yuille, and Cihang Xie. Are transformers more robust than cnns? *Advances in Neural Information Processing Systems*, 34:26831–26843, 2021.

[56] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12042–12051, 2022.

[57] Chongzhi Zhang, Mingyuan Zhang, Shanghang Zhang, Daisheng Jin, Qiang Zhou, Zhongang Cai, Haiyu Zhao, Xianglong Liu, and Ziwei Liu. Delving deep into the generalization of vision transformers under distribution shifts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7277–7286, 2022.

[58] Daquan Zhou, Zhiding Yu, Enze Xie, Chaowei Xiao, Animashree Anandkumar, Jiashi Feng, and Jose M Alvarez. Understanding the robustness in vision transformers. In *International Conference on Machine Learning*, pages 27378–27394. PMLR, 2022.

[59] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

[60] Yi Tay, Mostafa Dehghani, Samira Abnar, Hyung Won Chung, William Fedus, Jinfeng Rao, Sharan Narang, Vinh Q Tran, Dani Yogatama, and Donald Metzler. Scaling laws vs model architectures: How does inductive bias influence scaling? *arXiv preprint arXiv:2207.10551*, 2022.

[61] Sayak Paul and Pin-Yu Chen. Vision transformers are robust learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2071–2081, 2022.

[62] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12299–12310, 2021.

[63] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021.

[64] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[65] 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, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

[66] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[67] Tiange Xiang, Yongyi Liu, Alan L Yuille, Chaoyi Zhang, Weidong Cai, and Zongwei Zhou. In-painting radiography images for unsupervised anomaly detection. *arXiv preprint arXiv:2111.13495*, 2021.

[68] Fatemeh Haghighi, Mohammad Reza Hosseinzadeh Taher, Michael B Gotway, and Jianming Liang. Dira: Discriminative, restorative, and adversarial learning for self-supervised medical image analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20824–20834, 2022.

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

[70] 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.

[71] 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.

[72] Jiuwen Zhu, Yuexiang Li, Yifan Hu, Kai Ma, S Kevin Zhou, and Yefeng Zheng. Rubik's cube+: A self-supervised feature learning framework for 3d medical image analysis. *Medical Image Analysis*, 64:101746, 2020.

[73] Mark Chen, Alec Radford, Rewon Child, Jeff Wu, and Heewoo Jun. Generative pretraining from pixels. *Advances in Neural Information Processing Systems*, 2020.

[74] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9653–9663, June 2022.

[75] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022.

[76] Emmanuel J Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):1–37, 2011.

[77] Yiming Yang and William W Cohen. Singular value pruning of deep neural networks with application to dialogue response selection. *arXiv preprint arXiv:1509.08865*, 2015.

[78] Qingsong Hu, Deyu Zhang, Wei Zhang, and Xuelong Li. Truncated nuclear norm regularization for tensor completion. *arXiv preprint arXiv:1308.0737*, 2013.

[79] Chul Lee and Edmund Y Lam. Computationally efficient truncated nuclear norm minimization for high dynamic range imaging. *IEEE Transactions on Image Processing*, 25(9):4145–4157, 2016.

[80] Yao Hu, Zhongming Jin, Yi Shi, Debing Zhang, Deng Cai, and Xiaofei He. Large scale multi-class classification with truncated nuclear norm regularization. *Neurocomputing*, 148:310–317, 2015.

[81] Fanlong Zhang, Heyou Chang, Guowei Yang, Zhangjing Yang, and Minghua Wan. Truncated nuclear norm based low rank embedding. In *Biometric Recognition: 12th Chinese Conference, CCBR 2017, Shenzhen, China, October 28-29, 2017, Proceedings 12*, pages 708–715. Springer, 2017.

[82] Jinyu Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Low-rank tensor completion by truncated nuclear norm regularization. *arXiv preprint arXiv:1712.00704*, 2017.

[83] Xinyu Zhang, Guanghui Wang, Xiangzhao Li, Xuelin Liu, and Wei Liu. An efficient tensor completion method via truncated nuclear norm. *ScienceDirect*, 2020.

[84] Piotr Indyk, Ali Vakilian, and Yang Yuan. Learning-based low-rank approximations. In *Advances in Neural Information Processing Systems*, 2019.

[85] Ming Gao, Runmin Liu, and Jie Mao. Noise robustness low-rank learning algorithm for electroencephalogram signal classification. *Frontiers in Neuroscience*, 15:797378, 2021.

[86] Yuwu Lu, Zhihui Lai, Yong Xu, Xuelong Li, David Zhang, and Chun Yuan. Low-rank preserving projections. *IEEE transactions on cybernetics*, 46(8):1900–1913, 2015.

[87] Jiahuan Ren, Zhao Zhang, Richang Hong, Mingliang Xu, Haijun Zhang, Mingbo Zhao, and Meng Wang. Robust low-rank convolution network for image denoising. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6211–6219, 2022.

[88] Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks. *Trans. Mach. Learn. Res.*, 2023, 2023.

[89] Nadav Timor, Gal Vardi, and Ohad Shamir. Implicit regularization towards rank minimization in relu networks. In Shipra Agrawal and Francesco Orabona, editors, *International Conference on Algorithmic Learning Theory, February 20-23, 2023, Singapore*, volume 201 of *Proceedings of Machine Learning Research*, pages 1429–1459. PMLR, 2023.

[90] Maksym Andriushchenko, Dara Bahri, Hossein Mobahi, and Nicolas Flammarion. Sharpness-aware minimization leads to low-rank features. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[91] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[92] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[93] Mintong Kang, Yongyi Lu, Alan L Yuille, and Zongwei Zhou. Label-assemble: Leveraging multiple datasets with partial labels. *In Submission: Thirty-Sixth Conference on Neural Information Processing Systems*, 2021.

[94] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[95] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[96] Zhe Li, Chong Wang, Mei Han, Yuan Xue, Wei Wei, Li-Jia Li, and Li Fei-Fei. Thoracic disease identification and localization with limited supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8290–8299, 2018.

[97] Li Yao, Jordan Prosky, Eric Poblenz, Ben Covington, and Kevin Lyman. Weakly supervised medical diagnosis and localization from multiple resolutions. *arXiv preprint arXiv:1803.07703*, 2018.

https://doi.org/10.52202/079017-3719

[98] Yanbo Ma, Qiuhao Zhou, Xuesong Chen, Haihua Lu, and Yong Zhao. Multi-attention network for thoracic disease classification and localization. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1378–1382. IEEE, 2019.

[99] Yuxing Tang, Xiaosong Wang, Adam P Harrison, Le Lu, Jing Xiao, and Ronald M Summers. Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs. In *International Workshop on Machine Learning in Medical Imaging*, pages 249–258. Springer, 2018.

[100] Ivo M Baltruschat, Hannes Nickisch, Michael Grass, Tobias Knopp, and Axel Saalbach. Comparison of deep learning approaches for multi-label chest x-ray classification. *Scientific reports*, 9(1):1–10, 2019.

[101] Qingji Guan and Yaping Huang. Multi-label chest x-ray image classification via category-wise residual attention learning. *Pattern Recognition Letters*, 2018.

[102] Laleh Seyyed-Kalantari, Guanxiong Liu, Matthew McDermott, Irene Y Chen, and Marzyeh Ghassemi. Chexclusion: Fairness gaps in deep chest x-ray classifiers. In *BIOCOMPUTING 2021: Proceedings of the Pacific Symposium*, pages 232–243. World Scientific, 2020.

[103] Renato Hermoza, Gabriel Maicas, Jacinto C Nascimento, and Gustavo Carneiro. Region proposals for saliency map refinement for weakly-supervised disease localisation and classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 539–549. Springer, 2020.

[104] Eunji Kim, Siwon Kim, Minji Seo, and Sungroh Yoon. Xprotonet: Diagnosis in chest radiography with global and local explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15719–15728, June 2021.

[105] Fengbei Liu, Yu Tian, Yuanhong Chen, Yuyuan Liu, Vasileios Belagiannis, and Gustavo Carneiro. Acpl: Anti-curriculum pseudo-labelling for semi-supervised medical image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20706, 2022.

[106] Sina Taslimi, Soroush Taslimi, Nima Fathi, Mohammadreza Salehi, and Mohammad Hossein Rohban. Swinchex: Multi-label classification on chest x-ray images with transformers. *arXiv preprint arXiv:2206.04246*, 2022.

[107] Imane Allaouzi and Mohamed Ben Ahmed. A novel approach for multi-label chest x-ray classification of common thorax diseases. *IEEE Access*, 7:64279–64288, 2019.

[108] 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.

[109] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

[110] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

[111] Ross Wightman. Pytorch image models, 2019.

A Proofs

Proof of Theorem 3.1. It can be verified that at the t-th iteration of gradient descent for $t \geq 1$, we have

$$\mathbf{W}^{(t)} = \mathbf{W}^{(t-1)} - \frac{\eta}{n}\mathbf{F}^\top\left(\mathbf{F}\mathbf{W}^{(t-1)} - \mathbf{Y}\right). \tag{5}$$

https://doi.org/10.52202/079017-3719

It follows by (5) that

$$
\begin{aligned}
\mathbf{F}\mathbf{W}^{(t)} &= \mathbf{F}\mathbf{W}^{(t-1)} - \eta \mathbf{K}_n \left(\mathbf{F}\mathbf{W}^{(t-1)} - \mathbf{Y} \right) \\
&= \mathbf{F}\mathbf{W}^{(t-1)} - \eta \mathbf{K}_n \left(\mathbf{F}\mathbf{W}^{(t-1)} - \bar{\mathbf{Y}} \right), \qquad (6)
\end{aligned}
$$

where $\mathbf{K}_n = 1/n \cdot \mathbf{F}\mathbf{F}^\top$, $\bar{\mathbf{Y}} = \mathbf{U}^{(\bar{r})} {\mathbf{U}^{(\bar{r})}}^\top \mathbf{Y}$.

We define $\mathbf{F}(\mathbf{W}, t) \coloneqq \mathbf{F}\mathbf{W}^{(t)}$, then it follows by (6) that

$$
\mathbf{F}(\mathbf{W}, t) - \bar{\mathbf{Y}} = \left(\mathbf{I}_n - \eta \mathbf{K}_n \right) \left(\mathbf{F}(\mathbf{W}, t) - \bar{\mathbf{Y}} \right),
$$

which indicates that

$$
\begin{aligned}
\mathbf{F}(\mathbf{W}, t) - \bar{\mathbf{Y}} &= \left(\mathbf{I}_n - \eta \mathbf{K}_n \right)^t \left(\mathbf{F}(\mathbf{W}, 0) - \bar{\mathbf{Y}} \right) \\
&= - \left(\mathbf{I}_n - \eta \mathbf{K}_n \right)^t \bar{\mathbf{Y}},
\end{aligned}
$$

and

$$
\begin{aligned}
\left\| \mathbf{F}(\mathbf{W}, t) - \mathbf{Y} \right\|_{\mathrm{F}} &\le \left\| \mathbf{Y} - \bar{\mathbf{Y}} \right\|_{\mathrm{F}} + \left(1 - \eta \widehat{\lambda}_r \right)^t \left\| \bar{\mathbf{Y}} \right\|_{\mathrm{F}} \\
&\le \left\| \mathbf{Y} - \bar{\mathbf{Y}} \right\|_{\mathrm{F}} + \left(1 - \eta \widehat{\lambda}_r \right)^t \left\| \mathbf{Y} \right\|_{\mathrm{F}}. \qquad (7)
\end{aligned}
$$

As a result of (7), by using the proof of [17, Theorem 3.3,Corollary 6.7], for every $x > 0$, with probability at least $1 - \exp(-x)$,

$$
\begin{aligned}
L_{\mathcal{D}}(\mathrm{NN}_{\mathbf{W}}) \le{}& c_1 \left\| \mathbf{Y} - \bar{\mathbf{Y}} \right\|_{\mathrm{F}}^2 + c_1 \left(1 - \eta \widehat{\lambda}_r \right)^{2t} \left\| \mathbf{Y} \right\|_{\mathrm{F}}^2 \\
&+ c_2 \min_{h \in [0, r]} \left(\frac{h}{n} + \sqrt{ \frac{1}{n} \sum_{i=h+1}^{r} \widehat{\lambda}_i } \right) + \frac{c_3 x}{n}. \qquad (8)
\end{aligned}
$$

□

B More Experimental Results

B.1 NIH ChestX-ray14

Experimental setup. NIH ChestX-ray14 [10] consists of $112,120$ X-rays collected from $30,805$ unique patients. Each X-ray can have up to 14 associated labels, with the possibility of multiple labels per image. Following the official data split in [10], we use $75,312$ images for training and $25,596$ images for testing. The raw images from the dataset are sized 1024×1024. In our experiments, we scale down the input images to 224×224. We report the mean AUC (Area Under the Curve) for 14 distinct classes and conduct a comprehensive comparison with 18 widely recognized and influential baseline methods.

Results and Analysis. Table 6 presents the performance comparisons between several top-performing baseline models and their corresponding low-rank models on the NIH ChestX-ray14 dataset. Similar to Section 4.2, the ViTs are first pre-trained chest X-rays using Masked Autoencoders (MAE). Then, the pre-trained ViT-B network is fine-tuned on the NIH ChestX-ray14 dataset and achieves a mean AUC of 83.0. Next, we fine-tune ViT-B with low-rank regularization for another 75 epochs. The low-rank model, denoted as ViT-B-LR, achieves the new state-of-the-art performance with a mean AUC of 83.4. It is observed that all low-rank models achieve improvement in mean AUC compared to the corresponding base models. It is important to highlight that the research community dedicated four years to enhancing the AUC score for CNN-type architectures, advancing it from 74.5% to 82.2%, which was primarily attributed to the challenging nature of the classification with the NIH ChestX-ray14 dataset.

B.2 Comparison with A Comprehensive List of Baselines

We compare the results of LRFL models with a more comprehensive list of baselines on CheXpert. It is observed from the results in Table 7 that LRFL models significantly outperform all existing state-of-the-art methods on CheXpert.

Table 6: Performance comparisons between LRFL models and SOTA baselines on NIH ChestX-ray14. RN, DN, and SwinT represent ResNet, DenseNet, and Swin Transformer.

Method	Architecture	Pre-training	Rank	mAUC
Wang et al. [10]	RN50	ImageNet-1K	-	74.5
Li et al.[96]	RN50		-	75.5
Yao et al. [97]	RN&DN		-	76.1
Wang et al.[41]	R152		-	78.8
Ma et al.[98]	R101		-	79.4
Tang et al.[99]	RN50		-	80.3
Baltruschat et al.[100]	RN50		-	80.6
Guendel et al.[1]	DN121		-	80.7
Guan et al.[101]	DN121		-	81.6
Seyyed et al.[102]	DN121		-	81.2
Ma et al.[42]	DN121($\times$2)		-	81.7
Hermoza et al.[103]	DN121		-	82.1
Kim et al.[104]	DN121		-	82.2
Haghighi et al.[68]	DN121		-	81.7
Liu et al.[105]	DN121		-	81.8
Taslimi et al.[106]	SwinT		-	81.0
MoCo v2 [2]	DN121	X-rays (0.3M)	-	80.6
MAE [2]	DN121		-	81.2
RN-50 [2]	RN50	ImageNet-1K	-	81.8
RN-50-LR (Ours)	RN50		0.05r	82.2
DN-121 [2]	DN121	ImageNet-1K	-	82.0
DN-121-LR (Ours)	DN121		0.05r	82.4
ViT-S [2]	ViT-S/16	X-rays (0.3M)	-	82.3
ViT-S-LR (Ours)	ViT-S/16		0.05r	82.7
ViT-B [2]	ViT-B/16	X-rays (0.5M)	-	83.0
ViT-B-LR (Ours)	ViT-B/16		0.05r	**83.4**

Table 7: The table shows the performance of various state-of-the-art (SOTA) CNN-based and Transformer- based methods on CheXpert.

Method	Architecture	Rank	Atelectasis	Cardiomegaly	Consolidation	Edema	Effusion	mAUC (%)
Allaouzi et al.[107]	DN121	-	72.0	**88.0**	77.0	87.0	90.0	82.8
Irvin et al.[12]		-	81.8	82.8	93.8	93.4	92.8	88.9
Seyyedkalantari et al.[102]		-	81.2	83.0	90.0	88.3	93.8	87.3
Pham et al.[9]		-	82.5	85.5	93.7	93.0	92.3	89.4
Hosseinzadeh et al.[108]		-	-	-	-	-	-	87.1
Haghighi et al.[68]		-	-	-	-	-	-	87.6
Kang et al.[93]		-	82.1	85.9	**94.4**	89.2	93.6	89.0
DN121 (MoCo v2) [2]		-	78.5	77.9	92.5	92.8	92.7	88.7
DN121 [2]		-	81.5	77.6	89.4	92.3	92.0	88.7
ViT-S [2]	ViT-S/16	-	83.5	81.8	93.5	94.0	93.2	89.2
ViT-S-LR (Ours)	ViT-S/16	0.05r	**83.7**	86.3	90.9	93.7	93.1	89.6
ViT-B [2]	ViT-B/16	-	82.7	83.5	92.5	93.8	**94.1**	89.3
ViT-B-LR (Ours)	ViT-B/16	0.05r	81.6	85.4	93.4	**94.6**	93.9	**89.8**

B.3 Cross-Validation Results

The optimal values of the rank ratio γ, weighting parameter η, and learning rate μ decided by cross-validation for different models on different datasets are shown in Table 8.

Table 8: Optimal values of rank ratio γ, weighting parameter η, and learning rate μ decided by cross-validation for different models on different datasets.

Models	Parameters	NIH-ChestX-ray	COVIDx	CheXpert
ViT-S	γ	0.05	0.01	0.05
	η	5×10^{-4}	1×10^{-3}	1×10^{-3}
	μ	5×10^{-5}	2.5×10^{-5}	1×10^{-5}
ViT-B	γ	0.05	0.003	0.05
	η	5×10^{-4}	1×10^{-3}	1×10^{-3}
	μ	5×10^{-5}	2.5×10^{-5}	2.5×10^{-5}

In addition, the time for the entire cross-validation process in searching for the optimal values of the rank ratio γ, weighting parameter η, and learning rate μ are shown in Table 9. The evaluation is performed on 4 Nvidia A100 GPUs. As we use only 20% of the training data for cross-validation and train the models with each option for only 40% of the entire number of training epochs, the entire cross-validation process is efficient and does not largely increase the computation cost of the training process.

Table 9: Time Spent for cross-validation on NIH ChestX-ray14, CheXpert, and CovidX. All the results are reported in minutes.

Datasets	NIH ChestX-ray14	CheXpert	CovidX
ViT-S-LR	149	178	57
ViT-B-LR	172	285	69

B.4 Additional Ablation Study

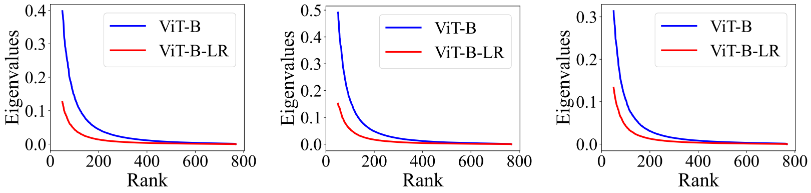
B.4.1 Study on the Kernel Eigenvalues and Kernel Complexity

Kernel complexity [17, 18, 19] is a widely-studied complexity measure for the generalization capability of kernel-based learning algorithms. In this section, we compare the eigenvalues of the kernel and kernel complexity of ViT-B-LR and ViT-B on ChestX-ray14, COVIDx, and CheXpert. Given the representations of all the training images $\mathbf{F}$ learned by ViT-B or ViT-B-LR, the kernel complexity of the gram matrix $\mathbf{K}_n = \frac{1}{n}\mathbf{F}\mathbf{F}^\top$, which is also defined in Section 3.3, can be computed by $\min_{h\in[0,n]}\left(\frac{h}{n}+\sqrt{\frac{\sum\limits_{i=h+1}^{n}\widehat{\lambda}_i}{n}}\right)$.

The eigenvalues of ViT-B-LR and ViT-B on ChestX-ray14, COVIDx, and CheXpert are illustrated in Figure 3. The computed kernel complexities of ViT-B-LR and ViT-B on ChestX-ray14, COVIDx, and CheXpert are shown in Table 10. It is observed that LRFL significantly reduces the kernel complexity of the image representations, which suggests that the LRFL models have lower generalization errors [17, 18, 19].

Table 10: Kernel complexity comparison between ViT-B-LR and ViT-B on ChestX-ray14, COVIDx, and CheXpert.

Method	ChestX-ray14		COVIDx		CheXpert	
	Kernel Complexity	h	Kernel Complexity	h	Kernel Complexity	h
ViT-B	0.0101	465	0.0207	303	0.0040	766
ViT-B-LR	0.0076	262	0.0155	187	0.0038	389



(a) ChestX-ray14 (b) COVIDx (c) CheXpert

Figure 3: Eigenvalues comparison between ViT-B-LR and ViT-B on ChestX-ray14, COVIDx, and CheXpert.

B.4.2 Experiments in Small Data Regimes

Experimental setup. We explore the effectiveness of low-rank features learned in scenarios with limited data availability, which is particularly significant given the challenges in acquiring high-quality data annotations in the medical imaging domain. We expect that LRFL models can demonstrate improved performance in such situations due to our theoretical guarantee of the better generalization capability of LRFL. We randomly select 5%, 10%, 15%, 20%, 25%, and 50% of the training data from the NIH ChestX-ray14 dataset and then fine-tune the base model using its default training configurations. We then train LRFL models for 20 epochs.

Results and analysis. As depicted in Table 11, our LRFL models consistently outperform their corresponding base methods across all data subsets, including 5%, 10%, 15%, 20%, 25%, and 50% on the NIH ChestX-ray14 dataset. Notably, the average improvement in performance is more substantial for the 5% data subset compared to the remaining subsets. For instance, ViT-B-LR exhibits a remarkable improvement of 1.05% for the 5% data subset, which significantly surpasses the improvements of 0.15%, 0.06%, 0.06%, 0.09%, and 0.11% observed for the 10%, 15%, 20%, 25%, and 50% training data subsets, respectively. These findings are consistent with our expectations, showcasing the strong generalization capability of LRFL models in mitigating over-fitting issues with limited data. In conclusion, our findings in the low-data regimes demonstrate the superiority of our LRFL in delivering more generalizable and robust representations for tasks with limited data availability, thereby contributing to the reduction of annotation costs.

Table 11: The table evaluates the performance of various models under low data regimes on the NIH ChestX-rays14 dataset. Models trained with low-rank features effectively combat overfitting in scenarios with limited data availability, thereby enhancing the quality of representations for downstream tasks.

Pre-training Dataset	Model	Label Fractions											
		5%		10%		15%		20%		25%		50%	
		Rank	mAUC	Rank	mAUC	Rank	mAUC	Rank	mAUC	Rank	mAUC	Rank	mAUC
X-rays(0.3M)	ViT-S	-	61.22	-	73.19	-	76.99	-	78.65	-	79.57	-	81.20
	ViT-S-LR(Ours)	0.05r	61.81	0.2r	73.84	0.04r	77.21	0.04r	78.86	0.05r	79.65	0.05r	81.35
X-rays(0.5M)	ViT-B	-	70.71	-	78.67	-	79.99	-	80.59	-	81.13	-	82.19
	ViT-B-LR (Ours)	0.05r	71.76	0.2r	78.82	0.2r	80.05	0.1r	80.65	0.05r	81.22	0.05r	82.30

B.4.3 Exploring Fine-tuning Strategies

Our LRFL method learns low-rank features by leveraging models pre-trained on the target dataset. In this section, we conduct an ablation study to investigate the significance of low-rank regularization in the fine-tuning process. A detailed comparative analysis of low-rank regularization against several performance-enhancing techniques, including mix-up [109], label smoothing [110], and EMA [111], is presented in Table 12. We performed an experiment by fine-tuning without low-rank regularization and other tricks, which serves as a baseline for studying the effects of fine-tuning strategies. All models underwent equivalent training epochs to ensure a fair comparison. The results demonstrate that LRFL models achieve the highest performance improvement compared to all other approaches. Notably, unlike natural images, applying mix-up, label smoothing, or EMA to the NIH ChestX-ray dataset leads to performance drops (see Table 12). Fine-tuning models pre-trained on the target dataset without low-rank regularization does not lead to performance improvements compared to fine-tuning with low-rank regularization. For example, the original ViT-S [2] achieves a mean AUC of 82.27% on NIH Chest Xray-14. Fine-tuning this model for 20 epochs without low-rank regularization leads to a mean AUC of 82.26%, whereas fine-tuning with low-rank regularization for 75 epochs results in a mean AUC of 83.40%. We observe similar results for all models based on low-rank features, demonstrating the significance of LRFL.

Table 12: Comparison of fine-tuning strategies on NIH ChestX-ray14.

Model	mAUC					
	Base Model	Fine-tuning	Mix-up [109]	Label Smoothing [110]	EMA [111]	LRFL
ViT-S	82.27	82.26	82.09	82.24	82.26	82.70
ViT-B	83.00	83.00	82.37	82.99	82.98	**83.40**

B.4.4 Additional Grad-CAM Visualization Results

Additional Grad-CAM visualization results of the Low-Rank ViT-Base on NIH ChestX-ray 14 are illustrated in Figure 5. Robust Grad-CAM visualization results of the Low-Rank ResNet-50 are illustrated in Figure 4. We visualize the parts in the input images that are responsible for the predictions of the ground-truth disease label for base models and low-rank models. The visualization results show that our low-rank models usually focus more on the areas inside the bounding box associated with the labeled disease. In contrast, the base models also focus on the areas outside the bounding box or even areas in the background.

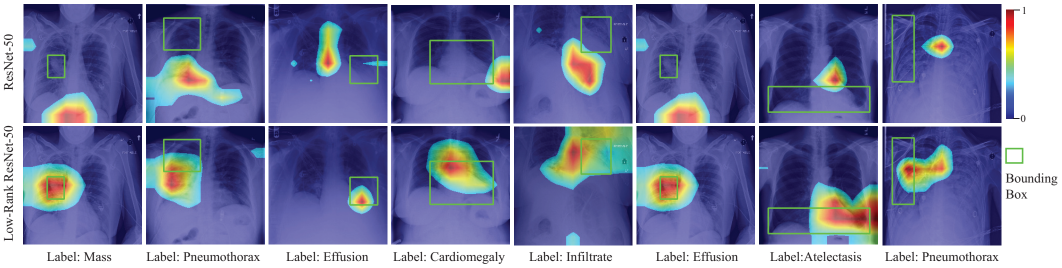


Figure 4: Robust Grad-CAM [95] visualization results on NIH ChestX-ray 14. The figures in the first row are the visualization results of ViT-Base, and the figures in the second row are the visualization results of Low-Rank ResNet-50.

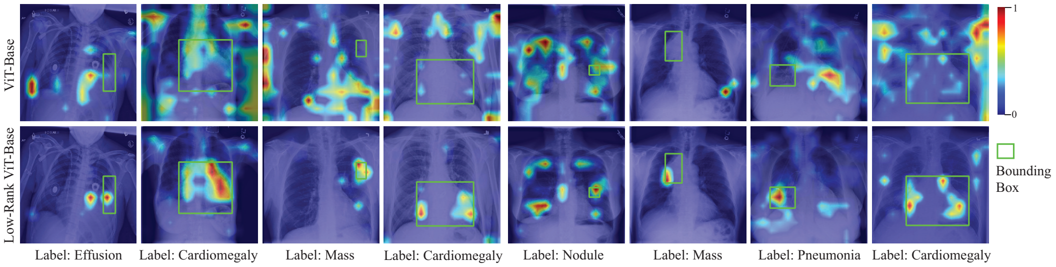


Figure 5: Grad-CAM visualization results on NIH ChestX-ray 14. The figures in the first row are the visualization results of ViT-Base, and the figures in the second row are the visualization results of Low-Rank ViT-Base.

B.4.5 Training Time Analysis

We evaluate the training time of LRFL models and compare them with the training time of the baseline models. The evaluation of LRFL models and baseline models is performed on 4 Nvidia A100 GPUs. It is observed from the results in Table 13 that the training time of LRFL models is comparable to the training time of LRFL models. The main computational overhead of LRFL models is the computation of the eigenvectors of the feature matrix **F** and the truncated nuclear norm. However, the computation overhead is largely reduced by avoiding performing SVD for the feature matrix **F** at every epoch, benefiting from the approximation algorithm we designed in Algorithm 1.

Table 13: Training time comparison between LRFL models and baseline models on NIH ChestX-ray14, CheXpert, and CovidX. All the results are reported in minutes.

Datasets	NIH ChestX-ray14	CheXpert	CovidX
ViT-S	54	90	23
ViT-S-LR	98	117	38
ViT-B	72	162	32
ViT-B-LR	113	185	45

C Training with Synthetic Data by Diffusion Models

In this section, we explore generative data augmentation using diffusion models. Section C.1 introduces the preliminaries of diffusion models and outlines the specific modifications made for our target task. In Section C.2, we discuss the implementation details of training of the diffusion model. Finally, we show some of the generated synthetic images in Figure 6.

C.1 Data Generation with the Diffusion Model

The diffusion model operates through a probabilistic framework, employing a forward noising process that gradually introduces noise to the original data $\mathbf{x}_0$. Initially, the model defines a distribution

$q(\mathbf{x}_t|\mathbf{x}_0)$ where $\mathbf{x}_t$ is progressively noised from $\mathbf{x}_0$ over time t. This distribution is governed by predetermined hyperparameters $\bar{\alpha}_t$, with $\mathbf{x}_t$ sampled using the reparameterization trick $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. As t advances, the noise contribution increases, leading $\mathbf{x}_t$ to become progressively noisier until it approximates a standard Gaussian distribution.

Following the training of the diffusion model, the focus shifts to the reverse process, aimed at denoising a noisy sample $\mathbf{x}_t$ to recover the original data $\mathbf{x}_0$. Utilizing a Gaussian noise $\mathbf{x}_T$ as the starting point, the model iteratively refines the sample using $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$, where μ_θ is parameterized to approximate the posterior mean of the forward process $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t))$, where $\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)\right)$. Training involves minimizing a simplified loss function L_{simple} to ensure accurate prediction of noise, facilitating effective denoising during the reverse process $L_{\text{simple}} = \mathbb{E}_{t,\mathbf{x}_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2\right]$.

We adopt a class of diffusion model known as the Diffusion Transformer (DiT) [94], chosen for its efficiency and token-agnostic conditioning, making it particularly suitable for our task. DiTs efficiently leverage label embedding for guidance and exhibit high compute efficiency, which is crucial for scaling to large datasets.

The Diffusion Transformer (DiT) model is trained on the CheXpert and COVIDx datasets, as described by [94]. The DiT model, specifically designed for text-based labels, operates without classifiers and instead relies on label embedding to guide the diffusion process. Modifications are made to the label embedding layer to adapt the model to the multi-label problem. Once the training is finished, images are sampled according to the label distribution of the original dataset to maintain the distribution of co-occurring diseases in the synthetic dataset.

C.2 Implementation Details

Training Settings of the Diffusion Model. Following the protocol in [94], the DiT is trained on 256×256 images for 2800 epochs, employing a global batch size of 512 distributed across 4 Nvidia A100 GPUs. Throughout training, a constant learning rate of 10^{-4} is maintained. After the training of the diffusion model is finished, synthetic images are sampled using a CFG scale of 4.0 and 128 sampling steps. To preserve the disease co-occurrence distribution within the synthetic dataset, identical image labels as those from the original dataset are utilized. The number of synthetic images added to the training set of each dataset is determined via cross-validation. We first generate synthetic images of the same size as the training set. The optimal percentage of synthetic images is selected using 5-fold cross-validation on the training data. Synthetic images are combined with the original dataset for further fine-tuning with low-rank regularization. Figure 6 presents examples of the synthetic chest X-rays generated using the aforementioned setting.

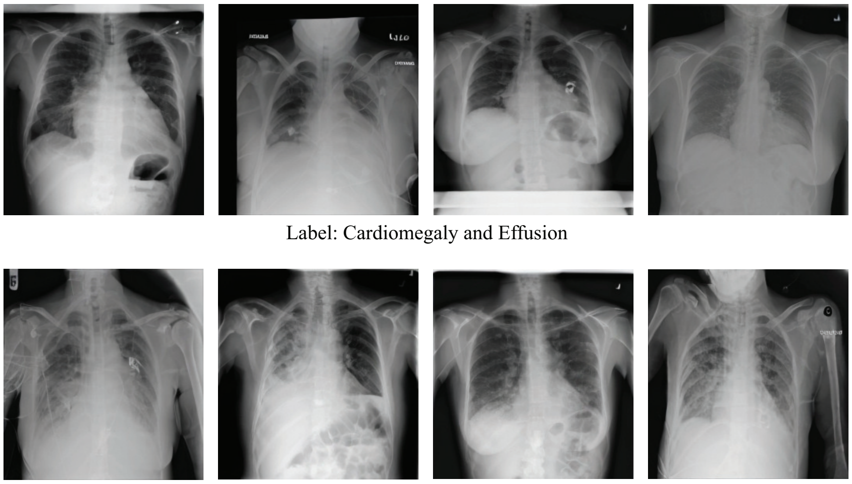
Table 14: Selected optimal percentage of images α on different datasets and models.

Dataset	CheXpert				COVIDx			
Models	ViT-S	ViT-S-LR	ViT-B-LR	ViT-B-LR	ViT-S	ViT-S-LR	ViT-B	ViT-B-LR
α	0.15	0.2	0.25	0.25	0.7	1.0	0.75	1.0

Tuning the Number of Synthetic Images n by Cross-Validation. We determine the optimal number of synthetic images for each dataset and the corresponding ViT variant. Let $N = \lceil \alpha \times n \rceil$, where α is the percentage of the images and n denotes the size of the training set of the target dataset. The values of α are selected through 5-fold cross-validation on the training data in each dataset. Specifically, α is chosen from the set $\{0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 1.0\}$. The optimal values of α selected by cross-validation for each dataset and ViT variant are presented in Table 14.

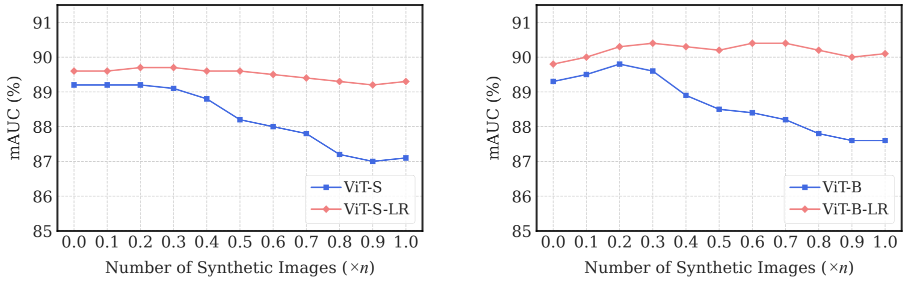
Training Settings of the LRFL Models with the Synthetic Images. Once we obtain the synthetic images generated by the diffusion model, we add the synthetic images into the training set of the target datasets, including COVIDx and CheXpert. We also leverage networks pre-trained on ImageNet [92] or chest X-rays [2] using Masked Autoencoders (MAE). The MAE pre-trained models are fine-tuned following the same pipeline as in Section 3.1 and the same implementation details as in Section 4.1.

https://doi.org/10.52202/079017-3719


Label: Cardiomegaly and Effusion

Label: Effusion, Infiltration and Pneumonia

Figure 6: Synthetic images generated using the Diffusion Model. The images in the first row are labeled Cardiomegaly and Effusion, and the images in the second row are labeled Effusion, Infiltration, and Pneumonia.



(a) ViT-S and ViT-S-LR

(b) ViT-B and ViT-B-LR

Figure 7: Performance comparisons between base models and LRFL models trained with different numbers of synthetic images added on CheXpert. n is the number of original training images in CheXpert.

C.3 Ablation Study on the Number of Synthetic Images

Although the usage of the generated synthetic images can improve the prediction accuracy of DNNs for image classification [13, 14, 15], too many synthetic images tend to introduce more noise to the augmented training data so excessive synthetic images can hurt the prediction accuracy of DNNs trained on the augmented training data [14]. Our proposed LRFL method, coupled with the selection of the amount of synthetic images, effectively mitigates this issue. In this section, we compare the performance of LRFL models with base models when different numbers of synthetic images are added to the training set of CheXpert. As illustrated in Figure 7, the performance of both the LRFL model and the base model can be initially improved with more synthetic images. However, after a certain point, even more, synthetic images start to hurt the performance due to the noise in the synthetic images, and the literature on using synthetic data for training classifiers such as [13] also has a similar observation. This is the reason why we need to perform a cross-validation on the size of the synthetic data for the best performance. Importantly, it can be observed that our LRFL models (ViT-S-LR or ViT-B-LR) usually improve the performance of the corresponding base models (ViT-S or ViT-B) on different choices of the size of the synthetic data. The improvements of our LRFL models over the corresponding base models tend to be more significant as the size of synthetic data increases. This observation justifies the effectiveness of LRFL in reducing the adverse effect of noise in the synthetic images. For example, ViT-B-LR outperforms ViT-B by 0.5% in mAUC when $0.1n$ synthetic images are added into the training set, and the improvement escalates to 2.5% with n synthetic images added into the training set where n is the size of the original training data.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction in Section 1 accurately reflect the paper's contributions and scope. The contributions of the paper are clearly stated in Section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [NA]

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes] .

https://doi.org/10.52202/079017-3719

Justification: The theorem and proposition in this paper are clearly stated in Section 3. The proof of the theorem is attached in Section A of the appendix of this paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The anonymous link to the code is attached at the end of the abstract. Implementation details for the experiments on image classification are stated in Section 4, and training the diffusion model is described in Section C of the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/079017-3719

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The anonymous link to the code is attached at the end of the abstract. Implementation details for the experiments on image classification are stated in Section 4, and training the diffusion model is described in Section C of the appendix. The datasets used in the experiments are public benchmarks. Detailed information on the datasets for image classification is stated in Section 4.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Experimental settings, including data splits, hyperparameters, type of optimizer, etc, for image classification are stated in Section 4, and training the diffusion model is described in Section C of the appendix. The datasets used in the experiments are public benchmarks.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We follow the manners in previous works [2, 94] for reporting the results on image classification, in Section 4, and for training of the diffusion model in Section C of the appendix.

Guidelines:

https://doi.org/10.52202/079017-3719

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computing resources used are specified in Section B.4.5 of our paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have read NeurIPS Code of Ethics in `https://neurips.cc/public/EthicsGuidelines` and confirm that the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

https://doi.org/10.52202/079017-3719

Justification: The experiments performed do not have societal impacts as we did not perform any experiments on real-world data collected from the society.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper poses no such risks for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper performs experiments on widely used public benchmarks, which are properly cited in Section 4 of our paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

https://doi.org/10.52202/079017-3719

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

https://doi.org/10.52202/079017-3719

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-3719