Journal of Privacy and Confidentiality
Vol. 15 (1) 2025

Submitted January 2024
Published March 2025

# DIFFERENTIALLY PRIVATE IMAGE CLASSIFICATION BY LEARNING PRIORS FROM RANDOM PROCESSES *

XINYU TANG †, ASHWINEE PANDA †, VIKASH SEHWAG, AND PRATEEK MITTAL

Princeton University, Princeton, NJ, USA
*e-mail address*: xinyut@princeton.edu

Princeton University, Princeton, NJ, USA
*e-mail address*: ashwinee@princeton.edu

Princeton University, Princeton, NJ, USA
*e-mail address*: vvikash@princeton.edu

Princeton University, Princeton, NJ, USA
*e-mail address*: pmittal@princeton.edu
*URL*: `https://github.com/inspire-group/DP-RandP`

Abstract. In privacy-preserving machine learning, differentially private stochastic gradient descent (DP-SGD) performs worse than SGD due to per-sample gradient clipping and noise addition. A recent focus in private learning research is improving the performance of DP-SGD on private data by incorporating priors that are learned from real-world public data. In this work, we explore how we can improve the privacy-utility trade-off of DP-SGD by learning priors from images generated by random processes and transferring these priors to private data. We propose DP-RandP, a three-phase approach, to combine the benefits of linear probing and full fine-tuning for the synthetic prior. We attain new state-of-the-art accuracy when training from scratch on CIFAR10, CIFAR100, MedMNIST, Camelyon17 and ImageNet for a range of privacy budgets $\varepsilon \in [1, 10]$. In particular, we improve the previous best reported accuracy on CIFAR10 from 60.6% to 72.3% for $\varepsilon = 1$.

## 1. Introduction

Machine learning models are susceptible to a range of attacks that exploit data leakage from trained models for objectives such as training data reconstruction and membership inference [Shokri et al., 2017, Balle et al., 2022]. Differential Privacy (DP) is the gold standard for quantifying privacy risks and providing provable guarantees against attacks [Dwork, 2008, Dwork and Roth, 2014]. DP implies that the outputs of an algorithm e.g., the final weights trained by stochastic gradient descent (SGD) do not change much (given by the privacy budget $\varepsilon$) across two neighboring datasets $D$ and $D'$ that differ in a single entry.

*Key words and phrases:* Differential Privacy, Image Classification.
* A preliminary version of this paper appeared in NeurIPS 2023 (Tang et al. [2023]).
† Equal contribution.


www.journalprivacyconfidentiality.org
DOI:10.29012/jpc.v0.i0.910

**Definition 1.1** (Differential Privacy)**.** A randomized mechanism $\mathcal{M}$ with domain $\mathcal{D}$ and range $\mathcal{R}$ preserves $(\varepsilon, \delta)$-differential privacy if for any two neighboring datasets $D, D' \in \mathcal{D}$ and for any subset $S \subseteq \mathcal{R}$ we have $\Pr[\mathcal{M}(D) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(D') \in S] + \delta$.

Differentially Private Stochastic Gradient Descent (DP-SGD) [Song et al., 2013, Abadi et al., 2016] is the standard privacy-preserving training algorithm for training neural networks on private data, with an update rule by $w^{(t+1)} = w^{(t)} - \frac{\eta_t}{|B|} \left( \sum_{i \in B} \frac{1}{c} \texttt{clip}_c(\nabla \ell(x_i, w^{(t)})) + \sigma \xi \right)$ where the changes to SGD are the per-sample gradient clipping $\texttt{clip}_c(\nabla \ell(x_i, w^{(t)})) = \frac{c \times \nabla \ell(x_i, w^{(t)})}{\max(c, ||\nabla \ell(x_i, w^{(t)})||_2)}$ and addition of noise sampled from a $d$-dimensional Gaussian distribution $\xi \sim \mathcal{N}(0, 1)$ with standard deviation $\sigma$. DP-SGD introduces bias and variance into SGD and therefore degrades utility, creating a challenging privacy-utility trade-off. For example, the state-of-the-art accuracy in private training is only 60.6% on CIFAR-10 at $\varepsilon = 1$ [Hölzl et al., 2023], while Dosovitskiy et al. [2021] obtains 99.5% accuracy non-privately.

Theoretical analysis of the DP-SGD update yields that noise addition is especially harmful to convergence at the start of training [Fang et al., 2023], and that pretraining on public data can greatly improve convergence in this initial phase of optimization [Li et al., 2022b] by providing a better initialization [Ganesh et al., 2023]. Previous works have improved the privacy-utility trade-off of DP-SGD by pretraining on large publicly available datasets, such as ImageNet [Deng et al., 2009], to learn visual priors [Mehta et al., 2023b,a, Panda et al., 2024, Bu et al., 2023b]. Other works assume that a small subset of in-distribution data is publicly available [Yu et al., 2021a, Amid et al., 2022, Li et al., 2022a, Nasr et al., 2023]. Despite the recent progress in public pretraining for private learning, Tramèr et al. [2024] raise the concern that pretraining on public datasets for private training may not be consistent with the privacy expectations of users. This is because this approach only protects the privacy of the fine-tuning dataset, not for the pretraining dataset. Web-scraped datasets like ImageNet, though publicly accessible, may also contain sensitive information from the individuals. In practice, these pretraining datasets also impose privacy concerns. For example, the ImageNet dataset contains sensitive content such as the face of individuals [Quach, 2019, Yang et al., 2022]. Directly pretraining models on such sensitive data without privacy considerations could lead to the privacy leakage of sensitive information. This intuitive concern is empirically validated by Abascal et al. [2024], who show that membership inference attacks can recover information about the pretraining data from the privately fine-tuned model.

Given these two competing objectives, the concern for the common "public" data for private learning and the desire for high-utility models, we seek to improve the utility for differentially private image classification without needing to access any real world images, even under the guise of using them as "public" data. Interestingly, previous work has uncovered that programmatically-generated synthetic data from random processes [Lee et al., 2001, Burton and Moorhead, 1987, Erhan et al., 2009, Karras et al., 2020] can be used in representation learning [He et al., 2020, Chen et al., 2020b,a] to learn highly useful visual priors [Kataoka et al., 2020, Baradad et al., 2021]. We visualize this synthetic data in Figure 1 and Figure 2, where we see that this synthetic data only contains general texture information and does not contain the aforementioned sensitive information often present in real world images. Therefore, training on these synthetic images does not incur the privacy cost of training on real-world datasets that may contain sensitive information from individuals. While prior works have shown that synthetic data can provide useful representations for

non-private learning [Lee et al., 2001, Burton and Moorhead, 1987, Erhan et al., 2009, Karras et al., 2020], using synthetic data for differentially private machine learning is underexplored.

For the ease of notation, we dub the setting of previous works [Mehta et al., 2023b,a, Panda et al., 2024, Bu et al., 2023b] as the public data setting, where some data from the real-world dataset is considered as non-private, and our setting as the synthetic data setting, where no real-world dataset is accessed without privacy protection. In this work, we ask: *How can we leverage synthetic data to improve differentially private image classification and how much utility improvement can we gain (compared to private learning without synthetic data and private learning with public data)?*

We leverage synthetic priors learned from programmatically generated images to boost the performance of DP training, and we realize their potential for a significant utility improvement for private training by designing a carefully calibrated training framework. In particular, we make the following key contributions:

- We provide empirical evidence that priors learned from random processes become more critical as the privacy budget decreases, i.e., the utility improvement with synthetic prior compared to without synthetic prior increases the privacy budget decreases (Figure 3), because priors provide fast convergence at the start of training (Figure 4). This is especially helpful for DP-SGD in private learning because in DP-SGD we need to pay privacy cost for every step. While for non-private training we can train for an arbitrary number of steps to get the best performance, for private learning we need to consider that training for longer under a fixed privacy budget requires adding more noise at each step. Therefore, the faster convergence that we derive as a benefit from synthetic prior has a much larger impact on private training than non-private training.
- We find that training a single linear layer (linear probing) on top of a pretrained feature extractor that has learned synthetic priors is more robust to large amounts of noise addition than end-to-end training of the entire network. We demonstrate this insight by linear probing with a small privacy budget $\varepsilon = 0.1$ to 57.1% on CIFAR10, achieving nontrivial performance with lower privacy cost than previous work has considered in the setting where no public data is accessed. We also observe that while linear probing from synthetic prior has diminishing returns on performance as the privacy budget increases, end-to-end training of the entire network continues to improve with large $\varepsilon$ but critically struggles for small privacy budgets.
- We harness our insights by proposing a privacy allocation strategy that combines the benefits of learning from priors, linear probing, and full training into our full method DP-RandP. Our proposed approach pretrains a feature extractor on synthetic data to learn priors from random processes without paying privacy cost, then pays a small privacy cost for linear probing and makes the best use of our remaining privacy budget by updating all parameters to adapt our learned features to the private data. Following the same intuition of DP-RandP to combine the benefits of linear probing and full fine-tuning, we propose an improved linear probing method for our synthetic prior that improves the performance of the direct linear probing, while remains simple and efficient.
- We evaluate DP-RandP across CIFAR10, CIFAR100, MedMNIST, Camelyon17 and the improved linear probing method across ImageNet, Camelyon17 against previous works and unlock new SOTA performance in the setting no public data accessed for all evaluated privacy budgets $\varepsilon \in [1, 10]$. We also discuss the comparison of our work with synthetic data and previous works on DP with public data. We note that there is still a noticeable gap between our work and those works that pretrain models on ImageNet as public data

and privately fine-tune on traditional benchmarks such as CIFAR10. Interestingly, for Camelyon17 as private data, we find that our method achieves performance comparable to that of the previous SOTA DP [Panda et al., 2024]; but crucially, Panda et al. [2024] assumes that ImageNet is public data, whereas we only need synthetic data. Camelyon17 is a dataset of medical images of histological lymph node sections labeled for the presence or absence of cancer, and therefore this is more similar to the practical scenarios with sensitive information. This demonstrates the potential efficacy of our method in practice, and we hope our work can provide insights for future work in improving the privacy-utility trade-off in private learning.

## 2. DP-RandP: Mitigating the effects of noise in DP-SGD with noise prior

In this section, we first give a brief motivation for why we consider a setting other than the common setting of the public data assumption, the introduction for the synthetic images, and the methods to encode synthetic prior from synthetic images. We then provide an initial exploration on how much synthetic prior benefits private training. Next we introduce our three-phase design DP-RandP that leverages the synthetic prior for differentially private image classification. We also introduce an improved linear probing baseline that is simple and efficient based on the similar intuition as DP-RandP.

### 2.1. Motivation.

**The potential concern of public data pretraining for private learning.** Recently, there is a line of works that study how to leverage public data for private learning [Mehta et al., 2023b,a, Panda et al., 2024, Bu et al., 2023b] (we discuss more related work for DP with public data in Section 4). Tramèr et al. [2024] raise the concern that the public data considered in such setting might also contain sensitive information. For example, the ImageNet dataset, while is publicly accessible and is considered as the public data in many previous DP with public data works [Mehta et al., 2023b,a, Panda et al., 2024, Bu et al., 2023b], still contains sensitive content such as the face of individuals [Quach, 2019, Yang et al., 2022]. Pretraining a model on a dataset like that without privacy protection will make the model to impose privacy leakage of such sensitive information [Meehan et al., 2023, Abadi et al., 2016]. In addition, Tramèr et al. [2024] also note that usually the "public pretraining then private fine-tuning" approach is evaluated on tasks that private fine-tuning data distribution is similar to of the public data distribution, and therefore it is hard to disentangle the improvements in general representation learning and improvements in private learning. We recommend interested readers to Tramèr et al. [2024] for a detailed discussion about these concerns for public pretraining.

Another line of work is how could we improve the privacy-utility trade-off without access to the public data. De et al. [2022] propose several techniques for private image classification by large batch size, making use of multiple data augmentations and exponential moving average (EMA), that are now standard techniques used in DP-SGD work [Sander et al., 2023]. Even with these techniques, the result in De et al. [2022] for CIFAR10 at $\varepsilon = 1$ is not better than the linear probing baseline on handcrafted features [Tramèr and Boneh, 2021]. This indicates that there is a potential improvement on the results in De et al. [2022]. Regarding the overlapping distribution issue between public data and private data, in addition to tasks that are widely benchmarked like CIFAR10 and ImageNet, we also consider medical tasks

including Camelyon17 that are more representative of the practical scenarios. We ablate our design choices on both kinds of tasks to understand the improvements of our work.

**Synthetic image generation without natural images.** Recent progress in computer vision shows that pretraining models on synthetic images without natural images [Baradad et al., 2021, 2022, Kataoka et al., 2020] can learn visual priors that are competitive to priors from natural images. The synthetic images can be generated either from texture and fractal-like noise [Baradad et al., 2022, Kataoka et al., 2020], or structure priors extracted from an untrained StyleGAN [Baradad et al., 2021]. Intuitively, training on such synthetic images does not incur any privacy cost as these synthetic images are generated without access to real-world images and are not linked to any real-world images with sensitive content. From the utility perspective, such synthetic images are also helpful in representation learning. These images carry several general image properties such as translation invariant property, and such general properties have shown improvements for differentially private image classification in Tramèr and Boneh [2021] by using ScatterNet [Oyallon and Mallat, 2015] to extract features.

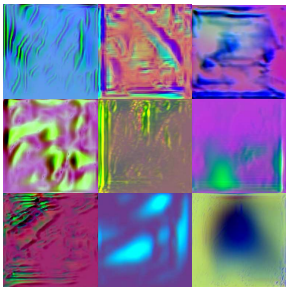

FIGURE 1. Example images from StyleGAN [Baradad et al., 2021].

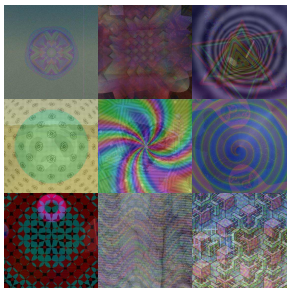

FIGURE 2. Example images from Shaders [Baradad et al., 2022].

**Synthetic prior from programmatically generated images for private training.** Motivated by the privacy property that real-world data is accessed and the potential utility improvement that is evidenced by Tramèr and Boneh [2021] for this synthetic data setting, we consider pretraining on synthetic data, generated by random processes [Baradad et al., 2021, 2022] (example images are in Fig. 1 and Fig. 2), as a *synthetic prior* in differentially private training. We use contrastive representation learning [Chen et al., 2020b,a, He et al., 2020, Wang and Isola, 2020], which aims to learn features invariant to common image transformations, to learn good visual features using synthetic images. At a high level, rather than using a random or 'cold' initialization for our downstream private dataset, we are using representation learning to obtain a 'warm initialization' that encodes priors learned from random processes. Across common natural vision tasks, synthetic priors also ensure that there is no privacy leakage of potential sensitive content into this 'warm initialization,' as no real-world data is accessed. In contrast, the 'warm initialization' with public data could leak sensitive information if the public data contains sensitive information [Abascal et al., 2024, Meehan et al., 2023].

2.2. **How much synthetic prior benefits private training? An initial exploration.** We obtain a warm initialization via representation learning on synthetic images and compare

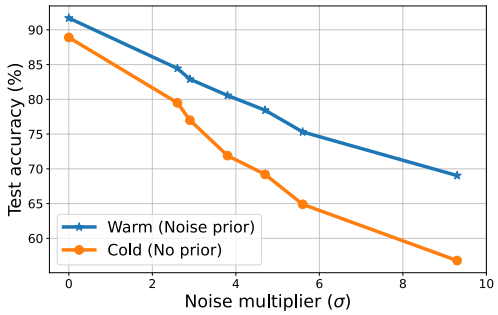


FIGURE 3. Comparison of training from a random initialization (cold) and pretrained encoder on a synthetic dataset (warm) across different privacy budgets (the x-axis is the corresponding $\sigma$).

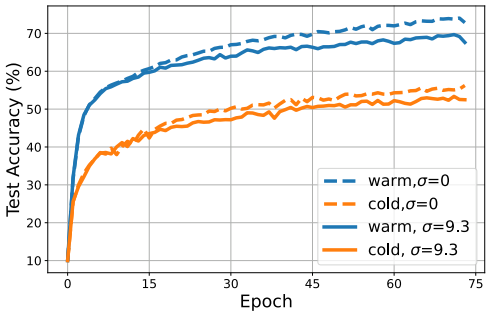


FIGURE 4. We zoom in on $\sigma = 0, \sigma = 9.3$ with learning rate= 0.4 and find that the improvement of the warm start over cold start is mostly due to the initialization using synthetic dataset.

it with a competitive DP-SGD baseline [De et al., 2022] that uses cold (random) initialization (Fig. 3). We find that while the warm initialization only improves performance by 2.7% when $\sigma = 0$, i.e., non-private training, the warm initialization improves performance by 12.5% when $\sigma = 9.3$ (equivalent to $\varepsilon = 1$). In Figure 4, we zoom in on a single point of comparison between $\sigma = 0$ (dashed) and $\sigma = 9.3$ (solid) for warm and cold initializations. We find that over the course of training, as more noise is added, both the $\sigma = 9.3$ warm and cold initializations not only converge at the same rate (given an appropriately small learning rate) but also diverge from the non-private runs at similar rates. *Although warm and cold start obtain different results in the private setting, the difference is mostly due to the initialization and is therefore magnified at smaller privacy budgets.*

We support this claim by drawing a connection to previous works. Ganesh et al. [2023] prove the existence of public datasets is necessary for private learning to achieve small test loss on target private datasets in their Theorem 2.1 for a small in-distribution public dataset and Theorem 2.2 for an out-of-distribution dataset.[1] Bu et al. [2023a] prove that in the gradient flow setting for the NTK regime, that is, when we are taking very small steps $\eta \to 0$, noise does not impact convergence. Mehta et al. [2023b], Panda et al. [2024] propose scaling the learning rate $\eta$ inversely with the noise $\sigma$. We combine our analysis with these previous works by noting that if we start training from fixed initialization and set the learning rate near-zero for small privacy budgets [Mehta et al., 2023b, Panda et al., 2024], we enter the regime [Bu et al., 2023a] where noise does not impact convergence. Additionally, achieving nontrivial performance for these small privacy budgets provides the first empirical evidence for the theory [Ganesh et al., 2023] that *only the initialization matters.*[2]

[1] Ganesh et al. [2023] show that for the out-of-distribution public data case, the public dataset and the private dataset are both necessary to achieve small test loss.

[2] Ganesh et al. [2023] consider a part of CINIC dataset as out-of-distribution data for CIFAR10. CINIC10 is a dataset that is very similar to CIFAR10, that is combined from 60,000 CIFAR10 images and 210,000 images from ImageNet with the same label class category as CIFAR10.

We gather our insights into a design goal that will enable us to achieve nontrivial performance under strict privacy constraints. We want to encode a learned prior into our initialization and then adapt this prior to private data. We now introduce DP-RandP, a method that achieves this key design goal.

### 2.3. **Our three-phase differentially private training framework.**

We propose a three-phase DP training framework DP-RandP that has two phases after pretraining on synthetic images, and significantly improves the performance of DP training (visualized in Fig. 5). We first learn synthetic priors by training a feature extractor on synthetic data (Phase-I). We then split our private training into 1) Learning the head classifier with frozen features (Phase-II) and 2) End-to-end training of the entire network to co-adapt the feature extractor and head classifier (Phase-III).

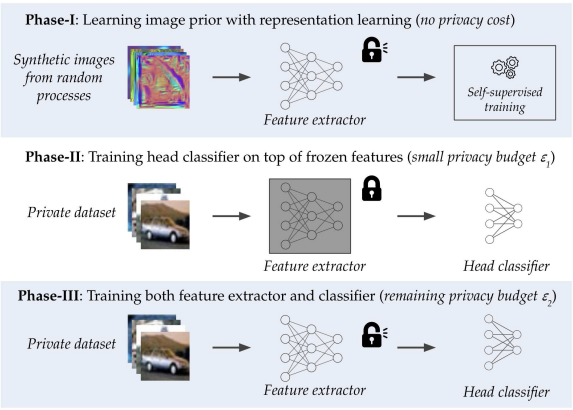


FIGURE 5. Our proposed DP training pipeline (DP-RandP) has three distinct phases. In Phase I, we sample images from random processes and train a feature extractor with representation learning to embed image priors beneficial for visual tasks. In Phase II, we spend a small privacy budget to train a linear classifier on top of extracted features of private data. In Phase III, we update all parameters with our remaining privacy budget. We demonstrate that incorporating image priors in Phase-I significantly improves DP training and adopting Phase II before training the whole network in Phase III can further improve test accuracy in DP training.

Our design is motivated by the strengths and weaknesses of linear probing and end-to-end training in the private setting. First, the $\ell_2$ norm of the Gaussian noise added to gradients in each DP-SGD step scales with the number of parameters in the model. After Phase I, the feature extractor network is already encoded with useful information from synthetic prior while the final head classification is randomly initialized. Also, as the head classifier typically has far fewer parameters than the feature extractor, updating this linear layer reduces the amount of added noise. Second, there is a large distribution shift between synthetic data from random processes and natural images found in private datasets. Linear probing merely inherits the frozen pre-trained features, but end-to-end training of all layers can improve the pre-trained features by adapting them to the private dataset. Because adapting to our private dataset may require many end-to-end training steps, we improve the convergence by

first learning a task-specific linear classifier that can be trained quickly, and then training the entire network end-to-end. DP-RandP satisfies our previously stated design goal by first obtaining a good initialization with pretraining, then transferring the prior encoded in this initialization to the private dataset with fast-converging linear probing, and finally end-to-end training to adapt to the private dataset as much as possible for a given privacy budget.

We further consider the design choice of how to allocate the privacy budget for the two private training phases ($\varepsilon_1$ and $\varepsilon_2$). Recall that our model is pre-trained on synthetic data in Phase I, thus it doesn't incur any privacy cost. In Phase II, we use DP-SGD with privacy budget $\varepsilon_1$ to train the linear classifier. We observe diminishing returns in accuracy with increasing $\varepsilon_1$, so we spend a smaller privacy budget on linear classifier training and allocate the rest to training the full network in Phase III.[3] Allocating a higher budget to full training does not have diminishing returns; consider that $\varepsilon_2 = \infty$, we recover non-private accuracy. We evaluate the performance of DP-RandP on the widely-used benchmark dataset CIFAR10 as well as Camelyon17 that is more aligned with practical use cases.

### 2.4. **An improved linear probing baseline motivated by DP-RandP.**

Our DP-RandP provides a promising direction for how to push the utility boundary from synthetic prior for differentially private image classification. There are potential other methods on how algorithm design could harness such utility gain. In addition to the utility gain, there could also be other motivations for developing algorithms to leverage such synthetic prior. For example, our DP-RandP involves a full fine-tuning stage. For a large scale dataset like ImageNet, training the full network would require significant computational resources (Sander et al. [2023] used 32 A100 GPUs). We leverage the core concept of DP-RandP and adapt it to a computationally efficient version, i.e., we train a linear layer with two additional modifications. We note that these two modifications are from previous works including Evci et al. [2022] and Sun et al. [2024], Wang et al. [2024]. We then reduce the computation cost for ImageNet on a single GPU to only a few hours.

Our first modification is to approximate full fine-tuning by linear probing on larger feature representations that we create by aggregating intermediate representations from the network. This is because each block of models like vision transformers learns a different representation [Evci et al., 2022]. However, linear probing only takes the representation from the penultimate layer as input and therefore the final representation may not be sufficient to learn the task. This modification is within the similar intuition in the full fine-tuning stage in DP-RandP to get better feature representations.

Our second modification is to approximate the work of a LayerNorm or other normalization layer that we would update during full fine-tuning of the entire ViT, by manually normalizing the features. Sun et al. [2024], Wang et al. [2024] find that such feature prepossessing could improve the privacy-utility trade-off in linear probing. To do this we first normalize each feature vector to a fixed norm. We next privately estimate the mean over the entire ImageNet feature vector dataset, using the Gaussian mechanism with a small privacy cost, and subtract the private mean from all feature vectors. This is equivalent to doing non-private centering and then adding the same Gaussian noise to the entire dataset. This

[3]We note that this additional stage in DP training introduces additional hyperparameter tuning. We include our hyperparameter choice in Appendix C and we note Phase II shares a similar hyperparameter choice across different $\varepsilon$s.

procedure can be thought of as a one-time approximation to the normalization layer, which is known to speed up training by centering the data. This also shares a similar intuition to Phase II in DP-RandP, which pays a small privacy cost to quickly adapt representations for the private images.

The investigation of this alternative method to our DP-RandP for linear probing with modifications is mainly motivated for computational efficiency for ImageNet tasks; *if we had enough compute to do full fine-tuning of the model on ImageNet with sufficiently large batch size, our original DP-RandP would still work.* These two modifications are simple but effectively improve the linear probing result. We demonstrate the effectiveness of this linear probing method on the ImageNet benchmark achieving new SOTA in the setting when no public data is available. We also present the results for Camelyon17. On Camelyon17, we also compare this method with other DP with public data works that use the linear probing of a pre-trained model that achieves SOTA results [Panda et al., 2024]. Interestingly, we find that our method achieves a comparable performance as Panda et al. [2024] on the Camelyon17 dataset.

Both of our methods based on synthetic prior improve previous works in the setting where no public data is available, which is promising. We note that there could be potential room in the algorithm design, which we will discuss in Section 4.

## 3. Evaluation

To outline the evaluation section we first overview our experimental setup and then evaluate the performance of DP-RandP on CIFAR10/CIFAR100/DermaMNIST/Camelyon17 in Section 3.2. We find that our method outperforms all previous works across multiple datasets, architectures, and privacy budgets. In Section 3.3, we find that both modifications in our linear probing method improve performance on ImageNet and Camelyon17. In Section 3.4 we find that our principled allocation of privacy budget $\varepsilon_1, \varepsilon_2$ between linear probing and end-to-end training is robust to different choices of $\varepsilon_1$ and $\varepsilon_2$. Next, in Section 3.5, we provide a quantitative comparison of DP-RandP and the improved linear probing results to DP with public data. We find that our private linear probing achieves comparable high performance as DP with public data on Camelyon17. Finally, we discuss the computational costs of our method and find that DP-RandP can provide computational savings over previous methods.

### 3.1. Experimental setup.

We give an overview of our experimental setup on datasets, models and implementation details.

**Learning priors from images generated by random processes with representation learning.** In Phase I we sample images from StyleGAN-Oriented [Baradad et al., 2021], and train a feature extractor on these synthetic datasets with representation learning [Chen et al., 2020b, Wang and Isola, 2020] with the loss function proposed in Wang and Isola [2020]. Although our method can accommodate any kind of synthetic data and representation learning method, we focus our evaluation on these datasets and methods. We consider other kinds of synthetic data and other representation learning in Appendix A.

**Datasets and models.** We evaluate DP-RandP on CIFAR10/CIFAR100 [Krizhevsky, 2009], DermaMNIST in MedMNIST [Yang et al., 2021, 2023], Camelyon17 [Bandi et al., 2018, Koh et al., 2021] and the private linear probing version of DP-RandP on ImageNet [Deng et al., 2009] and Camelyon17. For CIFAR10/CIFAR100, we use WRN-16-4 following De

et al. [2022]. We also use WRN-16-4 for Camelyon17. For MedMNIST, we use ResNet-9 following Hölzl et al. [2022]. We choose WRN-16-4 and ResNet-9 because these architectures for the corresponding datasets achieve the most compelling results in previous works De et al. [2022], Hölzl et al. [2022]. For linear probing on ImageNet and Camelyon17, we use a ViT-base [Dosovitskiy et al., 2021] feature extractor pretrained on Shaders-21k [Baradad et al., 2022] provided by Yu et al. [2024] and train a linear classifier. We provide the results on CIFAR10, CIFAR100, DermaMNIST, Camelyon17 in Section 3.2 and private linear probing results on ImageNet and Camelyon17 in Section 3.4. We also report results of WRN-40-4 on CIFAR10 in Appendix B.

**Implementation details.** To ensure a fair comparison with previous works [De et al., 2022, Sander et al., 2023, Hölzl et al., 2023], we use standard DP-SGD [Abadi et al., 2016] and make use of multiple data augmentations and exponential moving average (EMA) as proposed by De et al. [2022], that are now standard techniques used in DP-SGD work [Sander et al., 2023, Hölzl et al., 2023]. We allocate a small privacy budget $\varepsilon_1$ to Phase II and the remaining privacy budget $\varepsilon_2$ to Phase III according to the strategy detailed in Section 3.4. We report our results across different privacy costs and use $\delta = 10^{-5}$ for CIFAR10/CIFAR100/DermaMNIST, following previous works [De et al., 2022, Hölzl et al., 2022].[4] Following Ghalebikesabi et al. [2023], Lin et al. [2024], Liu et al. [2024], we consider $\varepsilon = 10$ and $\delta = 3 \times 10^{-6}$ for Camelyon17 dataset. When we report results, we report the standard deviation and accuracy averaged across 5 independent runs with different random seeds. We report implementation details for DP-RandP on CIFAR10/CIFAR100/DermaMNIST/Camelyon17 in Appendix C and improved private linear probing for ImageNet and Camelyon17 in Appendix G.

3.2. **Evaluation of DP-RandP.** We report the results of DP-RandP in Table 1, Table 2, Table 3, and Table 9 for CIFAR10, CIFAR100, DermaMNIST and Camelyon17 datasets, respectively.

**CIFAR10 results. DP-RandP outperforms all previous works across all privacy budgets.** In Table 1 we find that DP-RandP obtains higher performance than previous works [De et al., 2022, Hölzl et al., 2023, Tramèr and Boneh, 2021, Klause et al., 2022, Dörmann et al., 2021, Yu et al., 2021b, Papernot et al., 2021] on CIFAR10 across the standard evaluated privacy budgets $\varepsilon \in [1, 8]$.

We first compare DP-RandP to De et al. [2022]. Our CIFAR10 model optimizer and hyperparameter follow De et al. [2022]; the only difference is the use of Phase I and Phase II in DP-RandP to learn a prior from synthetic data and allocate a small privacy budget to linear probing. Crucially DP-RandP outperforms De et al. [2022] by *more than* 15% for the important privacy budget $\varepsilon = 1$.

Tramèr and Boneh [2021] use a ScatterNet [Oyallon and Mallat, 2015] to encode invariant image priors and Hölzl et al. [2023] use equivariant CNNs [Cohen et al., 2019] to learn transform invariant features. DP-RandP shares the same intuition of leveraging invariant image priors as Tramèr and Boneh [2021], Hölzl et al. [2023]. Instead of leveraging model architecture design for invariant features, we achieve this intuition by learning priors from

[4]We run experiments for $\varepsilon = \infty$ with per-sample gradient clipping but without noise, because we are interested in the ability of DP-RandP to mitigate variance. We note that there is accuracy degradation from the non-private baseline even at $\varepsilon = \infty$ due to the bias introduced by per-sample gradient clipping [Kamath et al., 2024]. When we report $\varepsilon = \infty$ results from previous work we mark them with $*$ when previous work does not report whether the non-private baseline uses clipping.

images generated from random processes and design our three-phase framework to optimize use of this prior. Although DP-RandP shares this intuition of leveraging invariant image priors [Tramèr and Boneh, 2021, Hölzl et al., 2023], DP-RandP achieves 12% improvement over Tramèr and Boneh [2021], Hölzl et al. [2023] who both achieve 60% at $\varepsilon = 1$. Our improvement comes both from leveraging the priors from synthetic images and our design of three learning phases that makes the best use of priors. We provide a detailed comparison to Tramèr and Boneh [2021] in Section 4 where we explain why and how the feature prior we learn from synthetic data provides better results than the feature prior provided by their architectures.

TABLE 1. Test accuracy (%) of DP-RandP and comparison to previous work on CIFAR10. Not shown in this table are Klause et al. [2022], Dörmann et al. [2021], Papernot et al. [2021], Yu et al. [2021b] because they achieve 71.5% at $\varepsilon = 7.5$ ,70.1% at $\varepsilon = 7.42$, 66.2% at $\varepsilon = 7.53$ and 63.4% at $\varepsilon = 8$ respectively, that are not on the pareto frontier of previous work. Note that Hölzl et al. [2023] use a variant of ResNet-9 by leveraging a equivariant CNN instead of WRN-16-4 that we use in this work.

| Method | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ | $\varepsilon = \infty$ |
|---|---|---|---|---|---|---|---|
| Tramèr and Boneh [2021] | 60.3 | 67.2 | 69.3 | – | – | – | 73.8* |
| De et al. [2022] | 56.8 | 64.9 | 69.2 | 71.9 | 77.0 | 79.5 | 88.9 |
| Hölzl et al. [2023] | 60.59 | 71.86 | 75.96 | 78.27 | 80.26 | 81.62 | – |
| DP-RandP | $72.32_{0.22}$ | $77.25_{0.07}$ | $79.99_{0.21}$ | $81.88_{0.27}$ | $84.01_{0.23}$ | $85.26_{0.11}$ | 91.69 |

**CIFAR100 results.** Previous work has not provided results on training from scratch for CIFAR100, perhaps because as we find in Table 2 the DP-SGD baseline following De et al. [2022] does not perform well for $\varepsilon = 3$. We believe this to be a competitive baseline, but DP-RandP outperforms it by more than 10% across $\varepsilon \in [3, 8]$. Although CIFAR10 and CIFAR100 are both benchmark computer vision datasets, we find CIFAR100 to be a much more challenging task for private learning. One possible explanation for this is that private classifiers struggle to distinguish between many classes because the added noise is more likely to shift the decision boundary. We use the same hyperparameter for CIFAR10 and CIFAR100, that are likely suboptimal for CIFAR100, and the performance gap between $\varepsilon = 8$ and $\varepsilon = \infty$ may be mitigated if we train on CIFAR100 for longer than we do on CIFAR10. We encourage the use of CIFAR100 as a standard benchmark for private learning in the future because we find limited room for improvement on CIFAR10. In particular, the private and non-private gap between $\varepsilon = 8$ and $\varepsilon = \infty$ for CIFAR10 is only $\approx 6\%$ for DP-RandP.

We have shown that DP-RandP outperforms previous work on the standard CV benchmarks of CIFAR10 and CIFAR100, and now consider the privacy sensitive medical datasets DermaMNIST and Camelyon17. Although CIFAR is a standard CV benchmark, there is limited previous work that evaluates privacy sensitive data in CV such as medical images.

**DermaMNIST results.** In Table 3 we find that DP-RandP achieve improvements from $\varepsilon = 1$ to $\varepsilon = 7.42$ by up to 2.78% over the DP-SGD baseline that we evaluate. Also, DP-RandP can achieve similar accuracy at $\varepsilon = 4$ as the result of DP-SGD at $\varepsilon = 7.42$, which reduces the privacy cost from $\varepsilon = 7.42$ to $\varepsilon = 4$. We also include the results of Hölzl et al.

TABLE 2. Test accuracy (%) of DP-RandP and comparison to DP-SGD baseline on CIFAR100.

| Method | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ | $\varepsilon = \infty$ |
|---|---|---|---|---|---|
| DP-SGD | 30.73 | 34.45 | 39.66 | 44.22 | 66.68 |
| DP-RandP | $43.33_{0.15}$ | $46.40_{0.31}$ | $51.53_{0.13}$ | $55.02_{0.21}$ | 71.68 |

[2022], that use equivariant neural networks [Cohen et al., 2019]. DP-RandP is a uniform framework that is applicable to any model architecture. We leave the systematic investigation of whether further utility improvement can be gained from applying the equivariant neural network and our framework for learning from priors to future work.

TABLE 3. We follow previous work [Hölzl et al., 2022] and report the validation accuracy (%) of DP-RandP on DermaMNIST. We also report the test accuracy in Appendix D.

| $\varepsilon$ | $\varepsilon = 1$ | $\varepsilon = 4$ | $\varepsilon = 7.42$ | $\varepsilon = \infty$ |
|---|---|---|---|---|
| baseline in Hölzl et al. [2022] | – | – | 72.41 | 78.48* |
| best in Hölzl et al. [2022] | – | – | 74.17 | 77.84* |
| DP-SGD we evaluated | $69.00_{0.37}$ | $71.78_{0.87}$ | $74.08_{0.41}$ | 77.27 |
| DP-RandP | $71.78_{0.40}$ | $74.82_{0.55}$ | $75.91_{0.29}$ | 79.26 |

TABLE 4. Test accuracy (%) of DP-RandP and comparison to DP-SGD and DP diffusion downstream classification accuracy on Camelyon17 for $(10, 3 \times 10^{-6})$-DP.

| Method | Accuracy |
|---|---|
| DPSDA [Lin et al., 2024] | 80.5 |
| DP-LDM [Liu et al., 2024] | 85.4 |
| DP-SGD in Ghalebikesabi et al. [2023] | 90.5 |
| DP Diffusion [Ghalebikesabi et al., 2023] | 91.1 |
| Our DP-SGD baseline | 84.09 |
| DP-RandP | 92.36 |

**Camelyon17 results.** We present the result of Camelyon17 in Table 4. Compared to the DP-SGD training from scratch baseline, our DP-RandP improves the performance by increasing the classification accuracy from 84% to 92%. We also compare our results to the line of works that either privately trains diffusion models or privately queries diffusion models to generate images, and then uses the generated images to train the downstream classification tasks non-privately [Ghalebikesabi et al., 2023, Lin et al., 2024, Liu et al., 2024]. These works also evaluate on the Camelyon17 dataset. Note that all of these three

works consider ImageNet as a public dateset and pre-train the diffusion model on ImageNet. These works generate data from the diffusion model satisfying the DP requirement and use Wide-ResNet-40-4 for downstream classification on the generated data. In addition, Ghalebikesabi et al. [2023] also pre-train the downstream classification model on ImageNet while other two works [Lin et al., 2024, Liu et al., 2024] do not. Among these three methods, Ghalebikesabi et al. [2023] update the full model parameters, Liu et al. [2024] only update the attention module in the latent diffusion models and Lin et al. [2024] freeze the diffusion model and privately select the generated images that are closer to private images via private evolution. Intuitively, Ghalebikesabi et al. [2023] have a better flexibility in adapting to the private domain and achieve the best performance among this three, and Lin et al. [2024] update no parameters in the model and yield accuracy lower than other two. This indicates that there is a large distribution shift between the private data Camelyon17 and public data ImageNet, which is more aligned with real-world privacy sensitive scenarios. The result of our work in Table 4 uses Wide-ResNet-16-4 without the ImageNet as an additional data source.[5] While our result in Table 4 use a smaller model for classification and we do not use ImageNet as public data, our method still outperforms these three methods. This result shows the effectiveness of the synthetic priors from Baradad et al. [2021, 2022] when compared with data generated from diffusion model in the DP way [Ghalebikesabi et al., 2023, Lin et al., 2024, Liu et al., 2024].

We find that the prior we learn from synthetic data is applicable to standard benchmarks and medical images. One direction for future work is to validate the robustness of this prior across more datasets. Our work also provides an initial study comparing the DP-SGD trained classification performance with general synthetic prior and non-private downstream classification training on DP generated data.

### 3.3. **Evaluation of the improved linear probing method.**

We now evaluate our improved linear probing method introduced in Section 2.4 on ImageNet and Camelyon17. We use a ViT-base [Dosovitskiy et al., 2021] model pretrained on Shaders-21k (the model checkpoint is provided by Yu et al. [2024]) for these two experiments. We also ablate the two modifications in this method on both tasks.

**ImageNet results.** We achieve new SOTA results on ImageNet. We achieve 39.39% accuracy at $\varepsilon = 8$ and the previous SOTA [Sander et al., 2023] is 39.2% at $\varepsilon = 8$. If we directly do Phase II with extracted features, we can achieve around 33% accuracy at $\varepsilon = 8$, that is comparable to De et al. [2022]. The direct linear probing result shows that linear probing (Phase II only after pretrained on synthetic data) is not enough for difficult tasks like ImageNet, and therefore updating full parameters in Phase III is necessary. As training the full ViT model on large datasets like ImageNet needs significant computational resources (for example, Sander et al. [2023] used 32 A100 GPUs), we run experiments on ImageNet by our improved linear probing method. Aggregating intermediate representations improves the performance from 33% to 37%. The feature normalization prepossessing, when combined with aggregating intermediate representation, improves performance from 37% to 39.39%, but provides minimal improvement on its own. We provide a detailed explanation and hyperparameter choice in Appendix G. This method is computationally efficient and

[5] De et al. [2022] show that the performance on CIFAR10 improves from Wide-ResNet-16-4 to Wide-ResNet-40-4, we observe a similar trend in our DP-RandP. We stay with Wide-ResNet-16-4 in this experiment as we use this model for our other main results.

one run can be done on a single A100 GPU in a few hours. We also provide the result for a stronger privacy guarantee, i.e., 26.54% at $\varepsilon = 1$.

TABLE 5. Test accuracy (%) of our private linear probing with additional designs on ImageNet.

| $\varepsilon$ | 1 | 8 |
|---|---|---|
| De et al. [2022] | - | 32.4 |
| Sander et al. [2023] | - | 39.2 |
| Ours (ViT) | 26.54 | 39.39 |
| (Std.) | 0.11 | 0.03 |

**Camelyon17 results.** After achieving the new SOTA and the validation of the two modifications of linear probing on ImageNet, we now evaluate this improved linear probing method on Camelyon17 dataset as this is more similar to the real-world scenarios. Direct linear probing from ViT achieves 90.46% on Camelyon17 at $(10, 3 \times 10^{-6})$-DP. Feature aggregation from intermediate layers improves the result to 92.22% and combining it with feature normalization as preprocessing we improve the result to 93.35%. Compared to the result in Table 4, this linear probing method achieves a better result. Table 4 uses a Wide-ResNet-16-4 and StyleGAN-Oriented as synthetic prior and here we use a VisionTransformer model and Shaders-21k as synthetic prior. This indicates that the improvements in architecture design and synthetic prior design could potentially improve the representation from synthetic prior for private learning and we leave it as future work. In addition, this improved linear probing method is a strong and efficient baseline, which could be used for future research. We also note that the DP linear probing by model pre-trained on ImageNet achieves comparable high performance as demonstrated in Panda et al. [2024] and we make comparison for these two linear probing methods in Section 3.5.

### 3.4. **Allocating privacy budget in DP-RandP.**

In this subsection we analyze the privacy budgets of Phase II ($\varepsilon_1$, linear probing) and Phase III ($\varepsilon_2$, full training) and how to allocate the overall privacy budget $\varepsilon$ among Phase II and Phase III. Due to the computational cost of DP-SGD, we only provide the ablation study of our DP-RandP on CIFAR10.

In Fig. 6 we observe that DP-RandP is robust to the key algorithmic design choice of how much privacy budget to allocate to Phase II and Phase III (Please check Appendix E for more details of Fig. 6 and more results on different $\varepsilon$).

In particular, we find that even the worst choice of $\varepsilon_1/\varepsilon$ provides results comparable to the previous SOTA on CIFAR10 at $\varepsilon = 1$.[6] We first investigate the behavior at each extrema and conclude a general allocation strategy that provides good performance across CIFAR10, CIFAR100, DermaMNIST and Camelyon17.

**Allocating the entire privacy budget to Phase II is competitive for small privacy budgets but provides diminishing returns.** This corresponds to the right extreme in Fig 6 ($\varepsilon_1/\varepsilon = 1$) and is equivalent to doing linear probing on top of extracted features. We follow the training recipe in Panda et al. [2024] and report the result of private

[6]Due to computational constraints we did not calculate error bars for Fig. 6, resulting in some non-convexity.

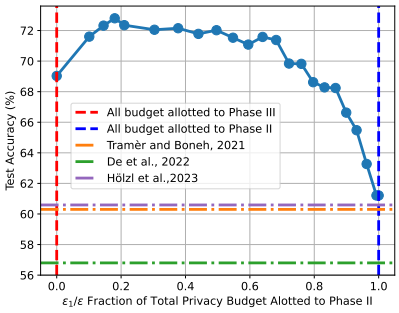


FIGURE 6. The fraction of total privacy budget allotted to Phase II for $\varepsilon = 1$. The performance is stable across a wide range of value [0.1, 0.4]. However, skipping either Phase-II or Phase-III leads to suboptimal test accuracy in DP training.

linear probing[7] in Table 6 for CIFAR10. We provide the detailed experimental set-up in Appendix F.

TABLE 6. Test accuracy (%) of our private linear probing and comparison to previous SOTA for private learning on top of extracted features [Tramèr and Boneh, 2021] on CIFAR10. Note that our result is training a linear layer on top of the extracted features. The result of previous SOTA is by training a CNN on top of extracted features. Tramèr and Boneh [2021] also report the private linear probing result 67.0% at $\varepsilon = 3$.

| $\varepsilon$ | 0.03 | 0.1 | 0.2 | 0.5 | 1 | 2 | 3 |
|---|---|---|---|---|---|---|---|
| SOTA | - | - | - | - | 60.3 | 67.2 | 69.3 |
| Ours | 40.64 | 57.10 | 60.89 | 65.10 | 67.78 | 69.92 | 71.08 |
| (Std.) | 2.59 | 0.42 | 0.26 | 0.21 | 0.17 | 0.09 | 0.14 |

We present the result of private linear probing on CIFAR10 by including more conservative privacy constraints like $\varepsilon = 0.03$. Our private linear probing can achieve 57.10% at $\varepsilon = 0.1$, that is comparable to the result of $\varepsilon = 1$ in De et al. [2022] that fully trains a WRN-16-4. Notably, previous work [Tramèr and Boneh, 2021] also trains a neural network on top of the handcrafted features using an untrained ScatterNet [Oyallon and Mallat, 2015]. DP-RandP is slightly better than their result at $\varepsilon = 3$. We can see that our non-private baseline (74.05%) is slightly better than the non-private baseline (73.8%) in Tramèr and Boneh [2021], that shows that our feature extractor is better than the ScatterNet and therefore improves the performance under DP-SGD. We include a detailed comparison to Tramèr and Boneh [2021] in Section 4. However, this private linear probing variant of DP-RandP can only achieve 74.05% with no noise added, that is lower than the result at $\varepsilon = 4$ in De et al. [2022], that shows that allocating all privacy cost to Phase II is a sub-optimal design choice.

[7]Note that here we are interested in the privacy allocation budget rather than pursuing the potential of synthetic prior, therefore we did not apply the two additional modifications described in Section 2.4.

**Allocating the entire privacy budget to Phase III struggles for small privacy budgets.** We report DP-RandP without Phase II on CIFAR10 in Table 7 (equals to $\varepsilon_1/\varepsilon = 0$ in Fig. 6). The result in Table 7 is slightly worse than DP-RandP in Table 1 and the utility gap between Table 7 and Table 1 decreases as $\varepsilon$ increases, which justifies the importance of Phase II in DP-RandP. Moreover, the result of DP-RandP without Phase II is significantly better than previous SOTA [De et al., 2022, Hölzl et al., 2023, Tramèr and Boneh, 2021], which shows that the feature extractor pretrained on images from random process can capture the image prior.

TABLE 7. Fully privately training a WRN-16-4 with warm initialization. Test accuracy on CIFAR10.

| $\varepsilon$ | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ | $\varepsilon = \infty$ |
|---|---|---|---|---|---|---|---|
| Accuracy(%) | $69.03_{0.23}$ | $75.31_{0.28}$ | $78.44_{0.19}$ | $80.56_{0.12}$ | $82.90_{0.10}$ | $84.45_{0.09}$ | 91.69 |

**A general privacy budget allocation strategy.** We have observed that allocating the entire privacy budget to linear probing or full training is suboptimal. We now propose a simple yet effective general strategy to allocate the privacy budget. For small $\varepsilon$ ($\varepsilon \ll 1$), we set $\varepsilon_1 = \varepsilon$ to allocate the entire privacy budget to Phase II. This is because allocating the entire privacy budget to linear probing is competitive for small privacy budgets. As $\varepsilon$ increases, we decrease $\varepsilon_1/\varepsilon$. This is because as $\varepsilon$ increases, the noise multiplier will decrease. Therefore, it is easier to train the linear probing layer as $\varepsilon$ increases, and the percentage of total steps allocated to Phase II can be reduced to train a good linear probing layer and we can use the remaining steps for Phase III. Because the closed-form computation of $\varepsilon$ with numerical privacy loss distribution accounting by Gopi et al. [2021] is challenging, we implement this strategy by using a fixed number of steps $n$ for linear probing in CIFAR10, CIFAR100, DermaMNIST, Camelyon17, and increasing the number of steps in Phase III as $\varepsilon$ increases. We present the privacy allocation $\varepsilon_1/\varepsilon$ on CIFAR10 in Appendix E for results in Table 1, which validates this strategy.

3.5. **In comparison to DP with public data.** A major direction in improving the privacy utility trade-off for DP-SGD is by incorporating priors that are learned on real-world public data [Yu et al., 2021a, Amid et al., 2022, Li et al., 2022a, Nasr et al., 2023, Mehta et al., 2023b,a, Panda et al., 2024, Pinto et al., 2024]. These priors are either from small in-distribution data [Yu et al., 2021a, Amid et al., 2022, Li et al., 2022a, Nasr et al., 2023] or large scale public data [Mehta et al., 2023b,a, Panda et al., 2024]. Pinto et al. [2024] use ImageNet as public data for pre-training and 10% of the private data as public data for dimension reduction.

We first present a quantitative comparison of DP-RandP and DP with public data works [Panda et al., 2024, Mehta et al., 2023b,a, Nasr et al., 2023, Pinto et al., 2024] in Table 8 on the common benchmarked dataset CIFAR10. Our DP-RandP is comparable with Nasr et al. [2023], which in fact utilizes a limited amount of in-distribution data as public data for pre-training. This indicates that the prior learned from images generated from random processes can help as much as the prior learned from limited in-distribution public data. Compared to works [Mehta et al., 2023b,a, Panda et al., 2024, Pinto et al., 2024] with access to large public data, there is still a gap between our DP-RandP and these

works. Specifically, the decreasing from $\varepsilon = 1$ to $\varepsilon = 0.1$, the performance drop in our work is much more significant than those results in Panda et al. [2024], Mehta et al. [2023a]. Note that we consider a different threat model to this line of work where we do not have access to public data [Tramèr et al., 2024] and instead could only utilize programmatically generated images. While Tramèr et al. [2024] point out the potential distribution overlap between ImageNet and CIFAR10, closing the gap between leveraging synthetic data and leveraging large-scale real public data is an interesting direction for future work.

TABLE 8. Comparison of DP-RandP with methods using public data. Test accuracy (%) on CIFAR10. Note that for $\varepsilon = 0.1$, the result of our work in this table is by allocating full privacy budget to Phase II in Table 6.

| Method | Model | Public Data | $\varepsilon = 0.1$ | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ |
|---|---|---|---|---|---|---|---|---|
| Nasr et al. [2023] | WRN-16-4 | 4% ID | – | 72.10 | 75.10 | 77.9 | 80.0 | – |
| Pinto et al. [2024] | ResNet50 | ImageNet+ 10% ID | 89.4 | 93.5 | 93.9 | – | – | – |
| Mehta et al. [2023b] | ViT-B/16 | ImageNet | – | 95.10 | 95.10 | 95.10 | – | 95.20 |
| Mehta et al. [2023a] | ViT-G/16 | JFT | 98.23 | 98.80 | 98.80 | 98.83 | – | 98.84 |
| Panda et al. [2024] | beitv2_large | ImageNet | 98.65 | 98.90 | – | – | – | – |
| DP-RandP | WRN-16-4 | Synthetic | 57.10* | 72.32 | 77.25 | 81.88 | 84.01 | 85.26 |

Panda et al. [2024] provide a strong linear probing result showing that when using ImageNet as public data for CIFAR10 as private data, there is only 0.1% degradation at $\varepsilon = 1$ compared to the non-private performance (99.00%). We now provide a comparison for linear probing on the model pre-trained on synthetic prior (our work) and the model pre-trained on ImageNet [Panda et al., 2024] on the Camelyon17 dataset. As Panda et al. [2024] show that the linear probing result could vary across different pre-trained models in their Table 4, we consider three models considered in Panda et al. [2024]. We present the comparison of our improved linear probing method and the method in Panda et al. [2024] on Camelyon17 in Table 9.

TABLE 9. Comparing the linear probing results of our work and Panda et al. [2024]. Test accuracy (%) on Camelyon17 dataset for $(10, 3 \times 10^{-6})$-DP.

| Method | Our method | Panda et al. [2024] | | |
|---|---|---|---|---|
| Prior | Synthetic Data | ImageNet | | |
| Model | ViT_base | ViT_base | ConvNeXt_xlarge | beitv2_large |
| # of params in feature extractor | 86M | 86M | 348M | 303M |
| Accuracy(%) | 93.35 | 93.27 | 93.26 | 94.32 |

Table 9 shows that for the similar model, i.e., the ViT-base model, our linear probing method, while only leveraging prior from synthetic data and no access to ImageNet, achieves comparable performance as ViT-base by Panda et al. [2024]. This indicates the potential efficacy of our method for sensitive information scenarios in practice. Even when compared to a larger feature extractor ConvNeXt_xlarge, our method is still comparable. We note that the beitv2_large that is used by Panda et al. [2024] for their main results also achieves a higher performance in Table 9, that demonstrates the impact of model architecture for private

learning. Our results on Camelyon17 provide a promising result that such synthetic prior could achieve comparable performance on prior from ImageNet for the healthcare-related image classification task. One limitation here is that we only make the comparison on a single dataset. Tramèr et al. [2024] call for *"researchers to create new benchmarks that more closely match envisioned deployments of private learning"* and *"choosing the right benchmarks for private machine learning is a consequential task, deserving of significant exploration and justification."* We anticipate future work in evaluating the utility improvement for the synthetic data prior and public data prior when new benchmark for private learning is available.

### 3.6. **Computational cost.**

DP-RandP consists of three phases. Some phases can be done in a single-run per dataset. For example, for feature extractor in Phase I, we only need to train a single feature extractor once for each evaluated dataset as the synthetic dataset and model architectures are the same. Also, for the private linear probing (LP). We report the computation cost in Table 10 for a single-run per dataset. For Phase II and Phase III, we need to go through these processes at each run of our training process. Also, after we get the extracted features for LP, we need to train a linear layer each time we run the experiments. We report the computation cost in Table 11 for these procedures.

De et al. [2022] also need to fully train a WRN-16-4 and the major additional computation cost for DP-RandP is Phase I compared to De et al. [2022]. However, the training in Phase I is done once for CIFAR10. Moreover, we can use the same feature extractor from Phase I for CIFAR10, CIFAR100 and Camelyon17.

TABLE 10. One-time per dataset computational cost on CIFAR10. These procedures only need to be done once for each evaluated dataset. Phase I can be shared for CIFAR10 /CIFAR100.

| | Phase I | feature extraction in LP |
|---|---|---|
| Time | 16 h | 1 min |

TABLE 11. Computational cost on CIFAR10 for hyperparameters given in Appendix C. These procedures are comparable to the standard training procedure such as De et al. [2022].

| | linear probing in LP | Phase II | Phase III |
|---|---|---|---|
| Time | 1 min | 12min | 5.5 h |

For the private linear probing experiment, each run of training a linear layer can be finished in 1 minute for CIFAR10 and 320 minutes for ImageNet while fully privately training a model costs much more time. A single run to privately train a WRN-16-4 for CIFAR10 takes around 5.5 hours for 875 steps with 1 A100 GPU in our evaluation. Also, as reported in previous work [Sander et al., 2023], a single run for ImageNet experiments needs to take four days using 32 A100 GPUs.

## 4. Discussion and related work

In this section we first provide a detailed comparison to previous work [Tramèr and Boneh, 2021] that also uses priors to improve DP-SGD image classification. We then give an overview of the broader body of work on improving the privacy utility trade-offs in DP-SGD. Finally, we discuss the previous work [Kumar et al., 2022] that also uses the two-stage training with domain-specific data for a different reasoning and also outline the potential improvements in algorithm design of our work.

**Discussion on DP-RandP and Tramèr and Boneh [2021].** Tramèr and Boneh [2021] find that training a neural network (linear layer or CNN) on top of 'handcrafted' ScatterNet [Oyallon and Mallat, 2015] features outperforms private 'deep' learning. While this method performs well for smaller values of $\varepsilon$, the non-private accuracy is limited because the features cannot be adapted to the private data. There are two key differences between DP-RandP and Tramèr and Boneh [2021]. In Phase I we use representation learning to train the feature extractor on images sampled from random processes, to learn the prior that extract transformations-invariant features. Our feature extraction process is therefore much more general, and while we explore the use of different synthetic data, model architectures, and representation learning methods, there are many more methods in each of these categories that we have not explored. The second difference is between the training on top of extracted features, that is used by Tramèr and Boneh [2021] and the private linear probing that we consider investigated in Table 6, and the combination of linear probing and full training that we use in DP-RandP. Comparing the improvements between DP-RandP and Tramèr and Boneh [2021] confirms that our empirical improvements are mostly due to the advantage of DP-RandP over DP linear probing. In particular, for $\varepsilon = 3$, exchanging the handcrafted features of Tramèr and Boneh [2021] for the pretrained feature extractor we use only improves performance by a modest $\sim 2\%$. However, our full DP-RandP exhibits more than 10% improvement. Our innovation over Tramèr and Boneh [2021] is therefore twofold: we introduce the potential of pretraining on synthetic data for the DP community, and also provide guidance on how to better transfer features learned from synthetic data to private training.

**DP with public data.** A major direction in improving the privacy utility trade-off in DP-SGD for image classification is the principled use of public data. Several works [Yu et al., 2021a, Amid et al., 2022, Li et al., 2022a, Nasr et al., 2023] make use of public data under a different threat model by treating a small fraction of the private training dataset as public. There is also another line of work that leverages a large real-world public dataset to pretrain models [Mehta et al., 2023b,a, Panda et al., 2024, Pinto et al., 2024].

Besides directly training image classification by DP-SGD, another direction is DP-trained generative models. The generated images can be used for classification tasks without additional privacy costs. Recent work [Ghalebikesabi et al., 2023] shows that DP diffusion models can achieve high-quality images when pretrained on large public data like ImageNet and achieve 88.8% classification accuracy for CIFAR10 at $\varepsilon = 10$.

Another line of work has shown the success of DP-SGD fine-tuning of pretrained large language models (LLMs) [He et al., 2023, Li et al., 2022c, Yu et al., 2022]. LLM pretraining can be framed as a way to learn structural priors from unstructured data [Brown et al., 2020].

**DP-SGD training from scratch.** In addition to the directly related works discussed above, the baselines for training from scratch that we compare to in this work are De et al.

[2022] and Sander et al. [2023]. De et al. [2022] make use of multiple techniques that are now a mainstay of DP-SGD training from scratch such as multiple data augmentations [Hoffer et al., 2020] that we also use. Sander et al. [2023] propose a method for estimating the best hyperparameters for DP training at scale using smaller-scale runs. Recent works [Hölzl et al., 2022, Hölzl et al., 2023] propose the use of inductive bias via architectural prior. They use DP-SGD to train the equivariant CNN architecture [Cohen et al., 2019] and achieve 71.8% at $\varepsilon = 2$ on CIFAR-10. We note that the design space of novel architectures that are especially compatible with DP is rich and mostly unexplored, and our approach is compatible with any advancements in this domain. In particular, we could use the architecture of Hölzl et al. [2022], Hölzl et al. [2023] in DP-RandP. We provide an exploration study on DP-RandP with equivariant CNN architecture in Appendix B. The exploration study shows the equivariant CNN also benefits from our DP-RandP and can achieve 76.87% at $\varepsilon = 2$ while without the synthetic prior the result is 71.86% in Hölzl et al. [2023].

**Two-stage training with domain-specific data.** The two-stage training of first training the classifier head and then tuning all hyperparameters has also shown to be effective for out-of-distribution (OOD) tasks [Kumar et al., 2022]. Their reasoning is that if the full parameters are directly updated using the in-distribution training data, this may lead to the loss of some general features learned in the pretrained stage and result in utility drop on OOD data. Our task of DP image classification is different from the OOD task [Kumar et al., 2022] but still shares some common intuition. After Phase I, our feature extractor has learned some useful priors while the classifier head has a random initialization. If we directly update the full network, too much noise is added to the full network, which may distort features learned in Phase I and lead to suboptimal performance. We also conduct experiments and find that, without synthetic prior (therefore model is randomly initialized), the two-stage training pipeline would not significantly improve the performance compared to directly training the full parameters (See Appendix H).

**Potential utility improvements by using synthetic prior.** In this work, we investigate using synthetic prior from programmatically generated image to improve the utility in differentially private image classification. We propose a three-phase framework DP-RandP to leverage both the benefits of linear probing and full fine-tuning. We also provide the results on an improved linear probing baseline with similar intuitions from DP-RandP. We provide an experimental analysis of the effectiveness of the three-phase approach DP-RandP in Section 3.4 only on CIFAR10. Our improved linear probing method, which follows a similar intuition to DP-RandP by using intermediate feature representations and fast adapting the domain information via feature preprocessing with a small privacy cost, provides utility improvements both on ImageNet and Camelyon17. Interestingly, we find that on the Camelyon17 dataset, our improved private linear method, while without access to ImageNet, achieves comparable performance as previous SOTA work in the line of DP with public data [Panda et al., 2024]. This set-up is more similar to the practical scenarios with sensitive information and demonstrates the effectiveness of our method. These findings, along with several other results in this work, show that leveraging synthetic prior to improve private learning for image classification is a promising direction.

We note that the three-phase approach DP-RandP introduces additional hyperparameter tuning processes for Phase 2. Appendix C includes the hyperparameters we use for DP-RandP, where for each dataset with different $\varepsilon$ we use a similar hyperparameter set-up for Phase II. Panda et al. [2024] propose an hyperparameter tuning method for private learning and show that their method also works for the linear probing baseline in their Table 2 on

CIFAR10 (the corresponding setting in our work is Table 6). This indicates our DP-RandP could potentially benefit from hyperparameter tuning methods like Panda et al. [2024] for private learning, which we leave to future work.

This work provides two methods on how to effectively leverage the synthetic prior. How algorithm design could better harness such utility gain is still an open-problem. For example, analysis on how the representation is learned in different layers could provide a better understanding for how to allocate the privacy budget accordingly. Such representation understanding could also be potentially helpful explaining why our methods achieve better performance than those images generated from diffusion models in a DP manner [Ghalebikesabi et al., 2023, Lin et al., 2024, Liu et al., 2024] on Camelyon17. Besides, there is a performance advantage when combined with more advanced models, i.e., ViT-base vs. Wide-ResNet-16-4, therefore the advancement in model architecture design could also provide benefits in this direction.

## 5. Conclusion

We leverage images generated from random processes and propose a three-phase training DP-RandP to optimize the use of synthetic prior. The evaluation across multiple datasets including benchmark datasets CIFAR10 and ImageNet, as well as medical datasets Camelyon17 and DermaMNIST, shows that DP-RandP can improve the performance of DP-SGD. For example, DP-RandP improves the previous best reported accuracy on CIFAR10 from 60.6% to 72.3% at $\varepsilon = 1$ and we also achieve comparable performance with prior SOTA DP with public data [Panda et al., 2024] on the Camelyon17 dataset. DP-RandP is a general framework for different datasets, models, and representation learning methods. Future improvements of designs in each of these categories would potentially improve the performance of DP-RandP. DP-RandP makes use of priors from synthetic images. It would be interesting to study whether DP-RandP would improve the priors for DP with public data. Also, investigating the priors beyond image domains, e.g., language and speech tasks, for differentially private training would also be of great interest.

## Acknowledgments

This work was supported in part by the National Science Foundation under grant CNS-2131938, the ARL's Army Artificial Intelligence Innovation Institute (A2I2), Schmidt DataX award, and Princeton E-ffiliates Award.

## References

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016. doi: 10.1145/2976749.2978318.

John Abascal, Stanley Wu, Alina Oprea, and Jonathan Ullman. Tmi! finetuned models leak private information from their pretraining data. *Proceedings on Privacy Enhancing Technologies*, 2014(3):202–223, 2024. doi: 10.56553/popets-2024-0075.

Ehsan Amid, Arun Ganesh, Rajiv Mathews, Swaroop Ramaswamy, Shuang Song, Thomas Steinke, Thomas Steinke, Vinith M Suriyakumar, Om Thakkar, and Abhradeep Thakurta. Public data-assisted mirror descent for private model training. In *Proceedings of the 39th International Conference on Machine Learning*, pages 517–535. PMLR, 2022. URL https://proceedings.mlr.press/v162/amid22a.html.

Borja Balle and Yu-Xiang Wang. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *Proceedings of the 35th International Conference on Machine Learning*, pages 394–403. PMLR, 2018. URL https://proceedings.mlr.press/v80/balle18a.html.

Borja Balle, Giovanni Cherubin, and Jamie Hayes. Reconstructing training data with informed adversaries. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1138–1156, 2022. doi: 10.1109/SP46214.2022.9833677.

Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE transactions on medical imaging*, 38(2):550–560, 2018. doi: 10.1109/TMI.2018.2867350.

Manel Baradad, Jonas Wulff, Tongzhou Wang, Phillip Isola, and Antonio Torralba. Learning to see by looking at noise. In *Advances in Neural Information Processing Systems*, pages 2556–2569, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/14f2ebeab937ca128186e7ba876faef9-Paper.pdf.

Manel Baradad, Richard Chen, Jonas Wulff, Tongzhou Wang, Rogerio Feris, Antonio Torralba, and Phillip Isola. Procedural image programs for representation learning. In *Advances in Neural Information Processing Systems*, pages 6450–6462, 2022. URL https://papers.nips.cc/paper_files/paper/2022/file/2a25d9d873e9ae6d242c62e36f89ee3a-Paper-Conference.pdf.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, pages 1877–1901, 2020. URL https://papers.nips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Zhiqi Bu, Hua Wang, Zongyu Dai, and Qi Long. On the convergence and calibration of deep learning with differential privacy. *Transactions on Machine Learning Research*, 2023a. ISSN 2835-8856. URL https://openreview.net/forum?id=K0CAGgjYS1.

Zhiqi Bu, Yu-Xiang Wang, Sheng Zha, and George Karypis. Differentially private optimization on large model at small cost. In *Proceedings of the 40th International Conference on Machine Learning*, pages 3192–3218. PMLR, 2023b. URL https://proceedings.mlr.press/v202/bu23a.html.

Geoffrey J Burton and Ian R Moorhead. Color and spatial structure in natural scenes. *Applied optics*, 26(1):157–170, 1987. doi: 10.1364/AO.26.000157.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020a. URL

https://proceedings.mlr.press/v119/chen20j.html.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020b. doi: 10.48550/arXiv.2003.04297.

Christopher A. Choquette-Choo, Krishnamurthy Dj Dvijotham, Krishna Pillutla, Arun Ganesh, Thomas Steinke, and Abhradeep Guha Thakurta. Correlated noise provably beats independent noise for differentially private learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=xHmCdSArUC.

Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral CNN. In *Proceedings of the 36th International Conference on Machine Learning*, pages 1321–1330. PMLR, 2019. URL https://proceedings.mlr.press/v97/cohen19d.html.

Soham De, Leonard Berrada, Jamie Hayes, Samuel L Smith, and Borja Balle. Unlocking high-accuracy differentially private image classification through scale. *arXiv preprint arXiv:2204.13650*, 2022. doi: 10.48550/arXiv.2204.13650.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Jinshuo Dong, Aaron Roth, and Weijie Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019. doi: 10.48550/arXiv.1905.02383.

Friedrich Dörmann, Osvald Frisk, Lars Nørvang Andersen, and Christian Fischer Pedersen. Not all noise is accounted equally: How differentially private learning benefits from large sampling rates. In *2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2021. doi: 10.1109/MLSP52302.2021.9596307.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19, 2008. doi: 10.1007/978-3-540-79228-4_1.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014. doi: 10.1561/0400000042.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009. URL https://papers.baulab.info/papers/also/Erhan-2009.pdf.

Utku Evci, Vincent Dumoulin, Hugo Larochelle, and Michael C Mozer. Head2Toe: Utilizing intermediate representations for better transfer learning. In *Proceedings of the 39th International Conference on Machine Learning*, pages 6009–6033. PMLR, 2022. URL https://proceedings.mlr.press/v162/evci22a.html.

Huang Fang, Xiaoyun Li, Chenglin Fan, and Ping Li. Improved convergence of differential private SGD with gradient clipping. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=FRLswckPXQ5.

Arun Ganesh, Mahdi Haghifam, Milad Nasr, Sewoong Oh, Thomas Steinke, Om Thakkar, Abhradeep Guha Thakurta, and Lun Wang. Why is public pretraining necessary for private model training? In *Proceedings of the 40th International Conference on Machine Learning*, pages 10611–10627. PMLR, 2023. URL https://proceedings.mlr.press/v202/ganesh23a.html.

Sahra Ghalebikesabi, Leonard Berrada, Sven Gowal, Ira Ktena, Robert Stanforth, Jamie Hayes, Soham De, Samuel L. Smith, Olivia Wiles, and Borja Balle. Differentially private diffusion models generate useful synthetic images. *arXiv preprint arXiv:2302.13861*, 2023. doi: 10.48550/arXiv.2302.13861.

Sivakanth Gopi, Yin Tat Lee, and Lukas Wutschitz. Numerical composition of differential privacy. In *Advances in Neural Information Processing Systems*, pages 11631–11642, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/6097d8f3714205740f30debe1166744e-Paper.pdf.

Jiyan He, Xuechen Li, Da Yu, Huishuai Zhang, Janardhan Kulkarni, Yin Tat Lee, Arturs Backurs, Nenghai Yu, and Jiang Bian. Exploring the limits of differentially private deep learning with group-wise clipping. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ozeOclVGPeX.

Kaiming He, Ross Girshick, and Piotr Dollar. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4917–4926, 2019. doi: 10.1109/ICCV.2019.00502.

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 (CVPR)*, pages 9729–9738, 2020. doi: 10.1109/CVPR42600.2020.00975.

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 (CVPR)*, pages 16000–16009, 2022. doi: 10.1109/CVPR52688.2022.01553.

Elad Hoffer, Tal Ben-Nun, Itay Hubara, Niv Giladi, Torsten Hoefler, and Daniel Soudry. Augment your batch: Improving generalization through instance repetition. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8126–8135, 2020. doi: 10.1109/CVPR42600.2020.00815.

Florian A. Hölzl, Daniel Rueckert, and Georgios Kaissis. Equivariant differentially private deep learning: Why dp-sgd needs sparser models. AISec '23, page 11–22. ACM, 2023. doi: 10.1145/3605764.3623902.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Florian A. Hölzl, Daniel Rueckert, and Georgios Kaissis. Bridging the gap: Differentially private equivariant deep learning for medical image analysis. *arXiv preprint arXiv:2209.04338*, 2022. doi: 10.48550/arXiv.2209.04338.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 448–456. PMLR, 2015. URL https://proceedings.mlr.press/v37/ioffe15.html.

Gautam Kamath, Argyris Mouzakis, Matthew Regehr, Vikrant Singhal, Thomas Steinke, and Jonathan Ullman. A bias-accuracy-privacy trilemma for statistical estimation. *Journal of the American Statistical Association*, pages 1–23, 2024. doi: 10.1080/01621459.2024.2443275.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8110–8119, 2020. doi: 10.1109/CVPR42600.2020.00813.

Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without natural images. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2020. doi: 10.1007/s11263-021-01555-8.

Helena Klause, Alexander Ziller, Daniel Rueckert, Kerstin Hammernik, and Georgios Kaissis. Differentially private training of residual networks with scale normalisation. *arXiv preprint arXiv:2203.00324*, 2022. doi: 10.48550/arXiv.2203.00324.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021. URL https://proceedings.mlr.press/v139/koh21a.html.

Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=UYneFzXSJWh.

Ann B Lee, David Mumford, and Jinggang Huang. Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *International Journal of Computer Vision*, 41:35–59, 2001. doi: 10.1023/A:1011109015675.

Tian Li, Manzil Zaheer, Sashank Reddi, and Virginia Smith. Private adaptive optimization with side information. In *Proceedings of the 39th International Conference on Machine Learning*, pages 13086–13105. PMLR, 2022a. URL https://proceedings.mlr.press/v162/li22x.html.

Xuechen Li, Daogao Liu, Tatsunori Hashimoto, Huseyin A Inan, Janardhan Kulkarni, YinTat Lee, and Abhradeep Guha Thakurta. When does differentially private learning not suffer in high dimensions? In *Advances in Neural Information Processing Systems*, pages 28616–28630, 2022b. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b75ce884441c983f7357a312ffa02a3c-Paper-Conference.pdf.

Xuechen Li, Florian Tramèr, Percy Liang, and Tatsunori Hashimoto. Large language models can be strong differentially private learners. In *International Conference on Learning Representations*, 2022c. URL https://openreview.net/forum?id=bVuP3ltATMz.

Zinan Lin, Sivakanth Gopi, Janardhan Kulkarni, Harsha Nori, and Sergey Yekhanin. Differentially private synthetic data via foundation model APIs 1: Images. In *The*

*Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=YEhQs8POIo.

Michael F Liu, Saiyue Lyu, Margarita Vinaroz, and Mijung Park. Differentially private latent diffusion models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=AkdQ266kHj.

Wesley Maddox, Shuai Tang, Pablo Moreno, Andrew Gordon Wilson, and Andreas Damianou. Fast adaptation with linearized neural networks. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pages 2737–2745. PMLR, 2021. URL https://proceedings.mlr.press/v130/maddox21a.html.

Casey Meehan, Florian Bordes, Pascal Vincent, Kamalika Chaudhuri, and Chuan Guo. Do ssl models have déjà vu? a case of unintended memorization in self-supervised learning. *Advances in Neural Information Processing Systems*, 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/854b6ec839294bf332db0d86e2f83c3f-Paper-Conference.pdf.

Harsh Mehta, Walid Krichene, Abhradeep Guha Thakurta, Alexey Kurakin, and Ashok Cutkosky. Differentially private image classification from features. *Transactions on Machine Learning Research*, 2023a. ISSN 2835-8856. URL https://openreview.net/forum?id=Cj6pLclmwT.

Harsh Mehta, Abhradeep Guha Thakurta, Alexey Kurakin, and Ashok Cutkosky. Towards large scale transfer learning for differentially private image classification. *Transactions on Machine Learning Research*, 2023b. ISSN 2835-8856. URL https://openreview.net/forum?id=Uu8WwCFpQv.

Fangzhou Mu, Yingyu Liang, and Yin Li. Gradients as features for deep representation learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BkeoaeHKDS.

Milad Nasr, Saeed Mahloujifar, Xinyu Tang, Prateek Mittal, and Amir Houmansadr. Effectively using public data in privacy preserving machine learning. In *Proceedings of the 40th International Conference on Machine Learning*, pages 25718–25732. PMLR, 2023. URL https://proceedings.mlr.press/v202/nasr23a.html.

Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2865–2873, 2015. doi: 10.1109/CVPR.2015.7298904.

Ashwinee Panda, Xinyu Tang, Saeed Mahloujifar, Vikash Sehwag, and Prateek Mittal. A new linear scaling rule for private adaptive hyperparameter optimization. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/panda24a.html.

Nicolas Papernot, Abhradeep Thakurta, Shuang Song, Steve Chien, and Úlfar Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9312–9321, 2021. doi: 10.1609/aaai.v35i10.17123.

Francesco Pinto, Yaxi Hu, Fanny Yang, and Amartya Sanyal. Pillar: How to make semi-private learning more effective. In *2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 110–139. IEEE, 2024. doi: 10.1109/SaTML59370.2024.00014.

Katyanna Quach. Inside the 1TB ImageNet data set used to train the world's AI: Naked kids, drunken frat parties, porno stars, and more., 2019. URL https://www.theregister.com/2019/10/23/ai_dataset_imagenet_consent.

Tom Sander, Pierre Stock, and Alexandre Sablayrolles. TAN without a burn: Scaling laws of DP-SGD. In *Proceedings of the 40th International Conference on Machine Learning*, pages 29937–29949. PMLR, 2023. URL https://proceedings.mlr.press/v202/sander23b.html.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2017. doi: 10.1109/SP.2017.41.

Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248, 2013. doi: 10.1109/GlobalSIP.2013.6736861.

Ziteng Sun, Ananda Theertha Suresh, and Aditya Krishna Menon. The importance of feature preprocessing for differentially private linear optimization. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=XlTDBZFXWp.

Xinyu Tang, Ashwinee Panda, Vikash Sehwag, and Prateek Mittal. Differentially private image classification by learning priors from random processes. In *Advances in Neural Information Processing Systems*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7058bc192a37f5e5a57398887b05f6f6-Paper-Conference.pdf.

Florian Tramèr and Dan Boneh. Differentially private learning needs better features (or much more data). In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YTWGvpFOQD-.

Florian Tramèr, Gautam Kamath, and Nicholas Carlini. Position: Considerations for differentially private learning with large-scale public pretraining. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/tramer24a.html.

Chendi Wang, Yuqing Zhu, Weijie J Su, and Yu-Xiang Wang. Neural collapse meets differential privacy: Curious behaviors of noisyGD with near-perfect representation learning. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/wang24cu.html.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020. URL https://proceedings.mlr.press/v119/wang20k.html.

Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 191–195, 2021. doi: 10.1109/ISBI48211.2021.9434062.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023. doi: 10.1038/s41597-022-01721-8.

Kaiyu Yang, Jacqueline H Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in imagenet. In *Proceedings of the 39th International Conference on Machine Learning*, pages 25313–25330. PMLR, 2022. URL https://proceedings.mlr.press/v162/yang22q.html.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in PyTorch.

*arXiv preprint arXiv:2109.12298*, 2021. doi: 10.48550/arXiv.2109.12298.

Da Yu, Huishuai Zhang, Wei Chen, and Tie-Yan Liu. Do not let privacy overbill utility: Gradient embedding perturbation for private learning. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=7aogOj_VYO0.

Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Large scale private learning via low-rank reparametrization. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12208–12218. PMLR, 2021b. URL https://proceedings.mlr.press/v139/yu21f.html.

Da Yu, Saurabh Naik, Arturs Backurs, Sivakanth Gopi, Huseyin A Inan, Gautam Kamath, Janardhan Kulkarni, Yin Tat Lee, Andre Manoel, Lukas Wutschitz, Sergey Yekhanin, and Huishuai Zhang. Differentially private fine-tuning of language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=Q42f0dfjECO.

Yaodong Yu, Maziar Sanjabi, Yi Ma, Kamalika Chaudhuri, and Chuan Guo. Vip: A differentially private foundation model for computer vision. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/yu24k.html.

## Appendix A. Ablation study for Phase I

We present our main results by using the WRN-16-4 model pretrained on Style-GAN dataset by Baradad et al. [2021] with representation learning [Wang and Isola, 2020] for CIFAR10/CIFAR100/DermaMNIST. In this section, we provide more ablation study on the different synthetic data and different representation learning methods in Phase I. We use CIFAR10 for the evaluation.

A.1. **Results on different synthetic data.** A number of synthetic datasets have been proposed by prior work. Baradad et al. [2021] consider a family of synthetic datasets generated by random processes, and report that the best synthetic dataset in terms of downstream performance is generated by an untrained StyleGAN with a specific initialization, denoted as StyleGAN-Oriented. Based on this prior work, we also use StyleGAN-Oriented as our main synthetic dataset throughout the main body of the paper. We also considered the Shaders dataset proposed in Baradad et al. [2022] for our ImageNet experiments. In this subsection we consider different choices of synthetic datasets and find that multiple synthetic datasets can provide good performance. That is, our results and analysis extend beyond StyleGAN-Oriented and can be applied to future proposed synthetic datasets.

We first use the feature extractor checkpoint[8] provided by Baradad et al. [2021] pretrained on different synthetic datasets (Please refer Baradad et al. [2021] for the full description of these datasets).

Note that Baradad et al. [2021] use a small AlexNet [Krizhevsky et al., 2012] as a feature extractor, which includes BatchNorm [Ioffe and Szegedy, 2015]. We freeze the feature encoder and only train the last linear model, therefore the feature extractor does not break our differential privacy guarantee. When we use the feature extractor in the main body we use WideResNet without BatchNorm.

Table 12 summarizes the results on different synthetic datasets. As in Baradad et al. [2021], the best synthetic dataset is StyleGAN-Oriented. However, even the worst-performing synthetic dataset (Dead leaves) performs similarly to the best prior work Tramèr and Boneh [2021], De et al. [2022]. Table 12 suggests that one potential future direction of DP-RandP is better synthetic data.

Table 12. Test accuracy (%) of private linear probing ($\varepsilon = 1$, training for 100 steps, that is the same as with WRN-16-4) on a small AlexNet trained on different synthetic images. StyleGAN-Oriented achieves the best performance. Note that all these images are generated without access to real-world images. Evaluation on CIFAR10 task.

| | Dead leaves textures | Stat Color+WMM | Untrained StyleGAN Sparse | High freq | Oriented | Feature Vis Dead leaves |
|---|---|---|---|---|---|---|
| Accuracy | 59.92 | 64.59 | 64.34 | 64.08 | 67.79 | 59.32 |

[8] https://github.com/mbaradad/learning_with_noise.

A.2. **Results on different representation learning methods.** In Table 13, we compare the representation learning method of Wang and Isola [2020] (results already in the main body) to MoCo [He et al., 2020] (we use default hyperparameters in the official repository) for use in Phase I. For Phase II and III, we use the same hyperparameters as in Table 16.

Similar to the main results, DP-RandP achieves significant improvements compared to baseline [De et al., 2022]. We find that using either of contrastive learning methods [Wang and Isola, 2020, He et al., 2020] can achieve 72% accuracy at $\varepsilon = 1$. Also, Table 13 shows that DP-RandP consistently improves upon DP-RandP without Phase II when using either of Wang and Isola [2020] or He et al. [2020] in Phase I. We note that while DP-RandP is robust to the two representation learning choices for Phase I, there is a small gap between the two methods as $\varepsilon$ increases. This suggests a future direction for further improving our method with a principled choice of contrastive learning method for Phase I.

TABLE 13. Test accuracy (%) of different representation learning methods in Phase I. Evaluation on CIFAR10 task.

| Method for Phase I | Phases | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ | $\varepsilon = \infty$ |
|---|---|---|---|---|---|---|---|---|
| Wang and Isola [2020] | DP-RandP | 72.32 | 77.25 | 79.99 | 81.88 | 84.01 | 85.26 | 91.69 |
| | DP-RandP w/o Phase II | 69.03 | 75.31 | 78.44 | 80.56 | 82.96 | 84.45 | 91.69 |
| MoCo [He et al., 2020] | DP-RandP | 72.79 | 76.26 | 78.32 | 79.78 | 81.64 | 82.84 | 90.84 |
| | DP-RandP w/o Phase II | 69.48 | 73.82 | 76.60 | 78.89 | 80.56 | 82.64 | 90.84 |
| – | Phase III only (De et al. [2022]) | 56.8 | 64.9 | 69.2 | 71.9 | 71.0 | 79.5 | 88.9 |

## APPENDIX B. ABLATION STUDY ON MODEL ARCHITECTURE

We conduct ablation study on model architecture for DP-RandP by considering WRN-40-4 and Equivariant-ResNet-9.

B.1. **WRN-40-4.** We present DP-RandP on CIFAR10 with experiments with WRN-16-4 in the main body and here we also present the WRN-40-4 result on CIFAR10 in Table 14. The result has a similar trend as De et al. [2022], WRN-40-4 can achieve better utility with more parameters. For example, at $\varepsilon = 1$, WRN-40-4 has a 0.83% increase compared to WRN-16-4. However, training a WRN-40-4 model takes a longer time. Training a WRN-16-4 for 875 steps takes 5.5 hours while the same amount of steps would take 12 hours for WRN-40-4. Given the utility improvement is within 2% improvement by changing from WRN-16-4 to WRN-40-4, we use WRN-16-4 to demonstrate the effectiveness of DP-RandP for the main experiments.

B.2. **Equivariant-ResNet-9.** We provide the result on DP-RandP on CIFAR10 at $\varepsilon = 2$ with Equivariant-ResNet-9 [9] in Table 15. Our hyperparameter setup mostly follows Hölzl et al. [2023] and includes steps 1125, batch size 8196, augmentation factor 4, SGD with momentum 0 and no weight decay. We also include the reproduced result of Hölzl et al. [2023] as cold initialization in Tab 15. Compared to the baseline that is Equivariant-ResNet-9 with cold initialization, DP-RandP improves the performance to 76.86%.

[9] https://github.com/hlzl/equivariant

TABLE 14. Ablation study on WRN-16-4 and WRN-40-4 on CIFAR10.

| Method | Model | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ |
|---|---|---|---|---|---|---|---|
| DP-RandP w/o Phase II | WRN-16-4 | 69.03 | 75.31 | 78.44 | 80.56 | 82.90 | 84.45 |
| | WRN-40-4 | 69.45 | 76.63 | 79.64 | 82.20 | 84.51 | 85.57 |
| DP-RandP | WRN-16-4 | 72.32 | 77.25 | 79.99 | 81.88 | 84.01 | 85.26 |
| | WRN-40-4 | 73.15 | 77.53 | 80.93 | 82.83 | 85.17 | 86.12 |

TABLE 15. Test accuracy for DP-RandP with Equivariant-ResNet-9 on CIFAR10 at $\varepsilon = 2$.

| | $\varepsilon = 2$ |
|---|---|
| Equivariant-ResNet-9 (cold init.) | 71.93 |
| DP-RandP without Phase II | 76.49 |
| DP-RandP | 76.87 |

## APPENDIX C. EXPERIMENTAL DETAILS

We use the Opacus library [Yousefpour et al., 2021] for the DP-SGD implementation. Our experiments are based on the open-source code[10] of Sander et al. [2023] and Baradad et al. [2021]. We also provide our code. For the noise multiplier $\sigma$, given sampling rate and total step size, $\sigma$ is precomputed according to privacy loss distribution accounting as implemented in Gopi et al. [2021] with epsError= 0.01 and rounded up to the precision of 0.1 to ensure that we do not underestimate the privacy loss.

**Hyperparameters.** Table 16, 17 and 18 summarize the hyperparameters for DP-RandP on CIFAR10, CIFAR100 and DermaMNIST respectively. We use the same total steps and batch size for CIFAR10 by following De et al. [2022]. We also use the same hyperparameters of batch size and steps for CIFAR100. For DermaMNIST, we use batch size 1024 because Hölzl et al. [2022] follow Klause et al. [2022] and Klause et al. [2022] use batch size 1024.

TABLE 16. Hyperparameters for DP-RandP on CIFAR10. Batch size is 4096 Augmult is 16, SGD with momentum 0 and no weight decay.

| $\varepsilon$ | 1 | 2 | 3 | 4 | 6 | 8 |
|---|---|---|---|---|---|---|
| Total Steps | 875 | 1125 | 1593 | 1687 | 1843 | 2468 |
| $\sigma$ | 9.3 | 5.6 | 4.7 | 3.8 | 2.9 | 2.6 |
| Steps in Phase II | 96 | 96 | 96 | 96 | 96 | 96 |
| LR in Phase II | 15 | 15 | 15 | 15 | 15 | 15 |
| LR in Phase III | 0.4 | 1 | 1 | 1.2 | 1.2 | 1.6 |

[10] https://github.com/facebookresearch/tan and https://github.com/mbaradad/learning_with_noise.

TABLE 17. Hyperparameters for DP-RandP on CIFAR100. Batch size is 4096 and Augmult is 16, SGD with momentum 0 and no weight decay.

| $\varepsilon$ | 3 | 4 | 6 | 8 |
|---|---|---|---|---|
| Total Steps | 1593 | 1687 | 1843 | 2468 |
| $\sigma$ | 4.7 | 3.8 | 2.9 | 2.6 |
| Steps in Phase II | 96 | 96 | 96 | 96 |
| LR in Phase II | 25 | 25 | 25 | 25 |
| LR in Phase III | 1.4 | 2 | 2.2 | 1.8 |

TABLE 18. Hyperparameters for DP-RandP on DermaMNIST. Batch size is 1024 and Augmult is 16, SGD with momentum 0 and no weight decay.

| $\varepsilon$ | 1 | 4 | 7.42 |
|---|---|---|---|
| Total Steps | 600 | 800 | 800 |
| $\sigma$ | 13.6 | 4.6 | 2.8 |
| Steps in Phase II | 48 | 48 | 48 |
| LR in Phase II | 2 | 2 | 2.8 |
| LR in Phase III | 0.2 | 1 | 1.2 |

TABLE 19. Hyperparameters for DP-RandP on Camelyon17 in Section 3.2. Batch size is 2048 and Augmult is 16, SGD with momentum 0 and no weight decay. Total number of training samples is $302,436$.

| $\varepsilon$ | 10 |
|---|---|
| Total Steps | 1000 |
| $\sigma$ | 0.62 |
| Steps in Phase II | 30 |
| LR in Phase II | 5.0 |
| LR in Phase III | 1.0 |

## APPENDIX D. ADDITIONAL RESULTS ON DERMAMNIST

We follow Hölzl et al. [2022] and report the validation accuracy of DermaMNIST in Table 3. Here we also report the test accuracy in Table 20 and we can see DP-RandP outperforms the DP-SGD baseline.

## APPENDIX E. PRIVACY ALLOCATION METHOD

We give a general privacy budget allocation strategy in Section 3.4. In this section, we give a detailed description of our privacy allocation strategy, the privacy ratio for CIFAR10 main results, and more results on different $\varepsilon$.

TABLE 20. Test accuracy (%) of DP-RandP on DermaMNIST.

| Method | $\varepsilon = 1$ | $\varepsilon = 4$ | $\varepsilon = 7.42$ | $\varepsilon = \infty$ |
|---|---|---|---|---|
| DP-SGD we evaluated | $68.34_{0.28}$ | $71.08_{0.51}$ | $72.58_{0.19}$ | 76.16 |
| DP-RandP | $71.19_{0.28}$ | $73.68_{0.24}$ | $75.04_{0.25}$ | 79.70 |

**Our privacy allocation strategy**. Given a total number of steps $N$, we use the first $N_1$ steps to train the head classifier for Phase II, and use the remaining $N_2 = N - N_1$ steps to train the full network for Phase III. We follow Panda et al. [2024] (that suggests 100 steps for linear probing) and set $N_1 = 96$ steps (this is closest to 100 steps and equals 8 epochs as each epoch contains 12 steps) in our experiment. For the x-axis in Fig. 6, we use PLD accounting as implemented in Gopi et al. [2021] to calculate $\varepsilon_1$ by calculating the privacy cost of $N_1$ steps and get $\varepsilon_1/\varepsilon$ as the x-axis. Although it is known that $\varepsilon_!$ does not increase linearly with $N_1$, $\varepsilon_!$ is monotonically increasing with $N_1$ and therefore we can use this method to compute the privacy ratio of Phase II. The $N_1$ steps in Fig. 6 include [0, 12, 24, 36, 48, 96, 144, 192, 240, 288, 336, 384, 432, 480, 528, 576, 624, 672, 720, 768, 816, 864, 875] with $N = 875$.

**Privacy ratio for CIFAR10 main results**. We visualize our privacy allocation strategy on CIFAR10 in Fig. 7, that is consistent with this strategy.

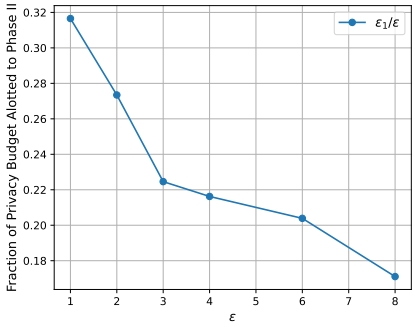


FIGURE 7. The fraction of total privacy budget allotted to Phase II ($\varepsilon_1/\varepsilon$) as a function of total privacy budget $\varepsilon$. As $\varepsilon$ increases, the ratio $\varepsilon_1/\varepsilon$ decreases.

**Additional experimental results on different** $\varepsilon$. We provide more results on different privacy allocations for different $\varepsilon$ from Table 21 to Table 25. Our main results in Table 1 in the main paper are produced by setting the number of epochs in Phase II to 8. From Table 21 to Table 25 we can see that our privacy splitting strategy is robust to several choices of the number of epochs in Phase II (e.g., 4, 8, 12, 16) in the evaluated privacy range. Furthermore, our privacy splitting strategy is better than allocating the entire privacy budget to either Phase II (rightmost) or Phase III only (leftmost).

TABLE 21. Test accuracy (%) for CIFAR10 $\varepsilon = 2$. 1125 steps. 94 epochs in total.

| Epochs for Phase II | 0 | 1 | 4 | 8 | 12 | 16 | 20 | 50 | 94 |
|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 75.30 | 76.31 | 77.03 | 77.35 | 76.74 | 77.05 | 77.10 | 75.06 | 63.10 |

TABLE 22. Test accuracy (%) for CIFAR10 $\varepsilon = 3$. 1593 steps. 133 epochs in total.

| Epochs for Phase II | 0 | 1 | 4 | 8 | 12 | 16 | 20 | 50 | 100 | 133 |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 78.34 | 79.20 | 79.71 | 79.74 | 79.48 | 80.27 | 80.04 | 79.21 | 75.47 | 63.93 |

TABLE 23. Test accuracy (%) for CIFAR10 $\varepsilon = 4$. 1687 steps. 141 epochs in total.

| Epochs for Phase II | 0 | 1 | 4 | 8 | 12 | 16 | 20 | 50 | 100 | 141 |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 80.63 | 81.24 | 81.76 | 81.46 | 81.61 | 82.01 | 81.56 | 81.36 | 78.07 | 64.41 |

TABLE 24. Test accuracy (%) for CIFAR10 $\varepsilon = 6$. 1843 steps. 154 epochs in total.

| Epochs for Phase II | 0 | 1 | 4 | 8 | 12 | 16 | 20 | 50 | 100 | 150 | 154 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 82.93 | 83.73 | 84.04 | 83.75 | 83.86 | 84.17 | 83.94 | 83.41 | 81.13 | 65.43 | 64.89 |

TABLE 25. Test accuracy (%) for CIFAR10 $\varepsilon = 8$. 2468 steps. 206 epochs in total.

| Epochs for Phase II | 0 | 1 | 4 | 8 | 12 | 16 | 20 | 50 | 100 | 150 | 200 | 206 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 84.37 | 84.84 | 85.43 | 85.17 | 85.21 | 85.37 | 85.08 | 85.08 | 83.84 | 82.74 | 69.17 | 64.85 |

## APPENDIX F. DETAILS FOR PRIVATE LINEAR PROBING ON CIFAR10

For CIFAR10, we follow Baradad et al. [2021] and use the alignment and uniformity loss proposed in Wang and Isola [2020] to pretrain a feature extractor WRN-16-4 on StyleGAN-Oriented dataset. Also we follow Baradad et al. [2021] and use the third to last layer and the dimension of this layer is 4096.

Note that, the right extreme in Fig. 6 is slightly higher than 60% at $\varepsilon = 1$, while the linear probing result in Table 6 is 67.78% at $\varepsilon = 1$. This is because representation learning [Wang and Isola, 2020, Chen et al., 2020a] usually adds additional representation layers for representation learning and may keep it for linear probing. As mentioned earlier,

we follow Baradad et al. [2021] and use the third to last layer and the dimension of this layer is 4096. For Fig. 6, we keep the exact same architecture as De et al. [2022] for a fair comparison where we did not use such embedding layers.

Table 26 summarizes the hyperparameter for DP-RandP without Phase III on CIFAR10 in Section 3.4.

TABLE 26. Hyperparameters for linear probing CIFAR10 experiments. Other hyperparameters include full batch size, SGD optimizer with momentum 0.9, 100 steps, no augmentation multiplicity.

| $\varepsilon$ | 0.1 | 0.2 | 0.5 | 1 | 2 | 3 | 4 | 6 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| $\sigma$ | 339 | 171 | 72 | 38 | 21 | 14 | 11 | 8 | 7 |
| LR | 0.8 | 2.2 | 5.5 | 10 | 20 | 40 | 40 | 60 | 60 |

## APPENDIX G. DETAILS FOR IMAGENET EXPERIMENTS

As discussed in Section 3.4, we achieve a new SOTA on ImageNet with additional designs for private linear probing. We first present the technical details of our method, which combines principled feature extraction and a private feature preprocessing approach. We then include the hyperparameters for our experiments on ImageNet.

G.1. **Method.** Our modifications on private linear probing include principled feature extraction and a private feature preprocessing approach. We qualify that neither of these steps is novel; feature extraction variants and feature preprocessing are very standard in feature extraction pipelines.

**Pretraining.** We use a pretrained ViT-base [Dosovitskiy et al., 2021] model by Yu et al. [2024], that is pretrained on the Shaders-21k dataset [Baradad et al., 2022] using MAE [He et al., 2022]. We use a different pretraining dataset here than for the simpler datasets, because models pretrained on Shaders-21k are observed to outperform those pretrained on StyleGAN [Baradad et al., 2022]. Furthermore, based on our initial experiments, models pretrained on StyleGAN with MoCo [He et al., 2020] do not perform well on ImageNet fine-tuning, only reaching $\approx 33\%$.

**Modifying the LP-FT recipe.** For the results on simpler datasets, we present a combination of linear probing and full fine-tuning (LP-FT [Kumar et al., 2022]). However, we note that for more complex datasets, it is more necessary to devote privacy budget to full fine-tuning because adapting the features learned from random priors to private datasets is more challenging. Based on the increase in the best values of $\varepsilon_1$ from CIFAR10 and CIFAR100, it seems likely that full DP fine-tuning is necessary to adapt the pretrained features to ImageNet. Unfortunately, we lack the computational resources to fine-tune ViT on ImageNet with DP as prior works [De et al., 2022, Sander et al., 2023] have noted, it may require hundreds or thousands of GPU-hours. Instead, we propose a *hybrid combination* of linear probing and full fine-tuning via principled feature extraction that is computationally efficient, running under 10 hours on a single A100, and also obtains better performance than the prior SOTA [Sander et al., 2023]. *We emphasize that these modifications are more*

*for computational efficiency; if we had enough compute to do full fine-tuning of the ViT on ImageNet with sufficiently large batch size, our original DP-RandP will still work.*

**Standard feature extraction.** Standard CNNs and ResNets iteratively refine the representation of the image at each layer, and the best representation of the image is produced at the penultimate layer. The head classifier at the end of the network learns a mapping between this representation and classes. Vision transformers are slightly different; each block learns a different representation. Nonetheless, the SOTA DP fine-tuning approaches that use pretrained ViTs as feature extractors still just use the representation from the penultimate layer as input to the linear layer [Panda et al., 2024]. When we use this approach for linear probing, our best result does not exceed 33.2%.

**Intuition behind principled feature extraction.** The intuition behind this approach is the same as the intuition in a long line of fine-tuning approaches [Maddox et al., 2021, Mu et al., 2020, Evci et al., 2022] and we do not claim any novelty for it. Given a sufficiently good pretrained initialization for the network [He et al., 2019], the domain adaptation should have low intrinsic rank such that the adaptation from the pretrained weights to the fine-tuning weights can be modeled in a lower-dimensional subspace than that of the full model parameters [Hu et al., 2022]. Results have shown that a linearized approximation of fine-tuning can obtain competitive performance. Therefore, for private learning in image classification, if we learn a linearization of the intermediate representations of the ViT, we can model the linearization of fine-tuning.

**Principled feature extraction.** The first challenge is the dimensionality of the intermediate representations. Although it may not seem large upon initial inspection, as ViTs may only have a representation size of 768, we actually want the representation before the pooling. The representation of the input after each block in the ViT has both a temporal and feature dimension, so we pool over the temporal dimension to gather a feature map of size (4, feature size). One alternative option here is actually to learn first what weights should be used for the final linear layer, as in Evci et al. [2022]. We can privatize this via the exponential mechanism. Initial results indicate this may be a very interesting direction for future work. However, this approach has a quite high computational cost, so we do not use it for the sake of reproducibility. Instead, we stride the average pooling such that the representations at each block are $4\times$ larger, and then we concatenate together the block-wise representations so that the final representation size is $4 \times$ num_blocks larger than it is in the standard feature extraction approach. For a ViT-base, num_blocks $= 12$ so this is $48\times$ larger for a final representation size and the feature size for each image is 36864.

**Feature normalization.** The first step in feature preprocessing is feature normalization. We normalize the feature vectors to a fixed norm of $C$ by the transformation below

$$x_i' = \frac{x_i \cdot C}{\|x_i\|_2}.$$

We treat $C$ as a fixed constant, and hence this normalization step doesn't result in any privacy loss about the dataset. We pick $C = 50$ because the representation multiplier from the principled feature extraction step is 48. Next we center the feature vectors around their mean $x_i = x_i - \frac{1}{|D|}\sum_{j\in D} x_j$, which requires private mean estimation. That is, the input to the training method will just be the difference between the feature and the noisy feature mean.

**Private mean estimation.** Now we introduce the motivation behind the feature normalization, so that we can do private mean estimation in high dimensions without

prohibitive error rates. Given that all the feature vectors have the same dimension and fixed norm, the optimal error rate for private mean estimation will be obtained via the Gaussian mechanism. That is, we first compute the true mean and then add Gaussian noise scaled to the $\ell_2$ sensitivity of the mean. The sensitivity of the mean is $C/N$; adding or removing any datapoint can change the $\ell_2$ sensitivity by at most $C$, and there are $N$ datapoints (for ImageNet, $N = 1281167$). We do a hyperparameter search here to find the best amount of the overall privacy budget to dedicate to this step, which we can do efficiently by saving the true mean and then adding noise for each value of $\varepsilon$ we consider. We find that a very noisy estimate is sufficient, because the noise is added to the mean vector and then used to normalize all the features such that all datapoints have the same noise. This correlated noise is highly tolerable for our approach, in line with concurrent work that indicates correlated noise has much less accuracy degradation [Choquette-Choo et al., 2024].

**Related work on private feature preprocessing.** One concurrent work [Sun et al., 2024] conducts a theoretical analysis that feature preprocessing provably reduces the error rate of DP linear regression and provides experiments when fine-tuning a model that is pretrained on ImageNet (that is, the regime of Table 8. Another concurrent work [Wang et al., 2024] conducts theoretical analysis that feature preprocessing provably reduces the error rate of DP-GD from extracted features by connecting to the neural collapse regime and provides experimental validation when fine-tuning a model that is pretrained on ImageNet. These concurrent works provide intuition behind the success of our private feature preprocessing, although we note that neither theoretical guarantee is actually applicable to our setting because we use an entirely different method (pretraining on synthetic data, principled feature extraction, and private feature preprocessing). Our method therefore validates the theory of these concurrent works by applying private feature preprocessing in conjunction with principled feature extraction to do private linear probing on one of the most challenging datasets for DP image classification, achieving a new SOTA for methods that do not use public data.

G.2. **Results.** We achieve 39.39% accuracy at $\varepsilon = 8$, and the previous SOTA [Sander et al., 2023] is 39.2% at $\varepsilon = 8$. Although this improvement is minor, we note that we have not been able to reproduce the results in Sander et al. [2023] using their provided code, who themselves were not able to reproduce the result from De et al. [2022] that they compare to. This reproducibility gap can be attributed to the high variance of DP training, which itself is difficult to mitigate because of the enormous computational cost required to get SOTA results on ImageNet privately. We hope that by providing our code we can bridge this gap.

We include hyperparameters of our private linear probing results on ImageNet in Table 27. In Table 27, we use $\sigma_1$ for the private mean estimation step and $\sigma_2$ for DP-SGD. Because we are using the full batch, we can use the composition theorem in Gaussian differential privacy [Dong et al., 2019] (Theorem 2.7 and Corollary 3.3) to compute the privacy loss. With $T$ steps in DP-SGD, our private linear probing is equivalent to a one-step Gaussian mechanism with noise multiplier $\sigma$, where

$$\sigma = \frac{1}{\sqrt{\frac{1}{\sigma_1^2} + \frac{T}{\sigma_2^2}}}$$

We can then compute $(\varepsilon, \delta)$-DP by computing the privacy curve of Gaussian mechanism [Balle and Wang, 2018, Dong et al., 2019].

TABLE 27. Hyperparameters for linear probing on ImageNet. $\sigma_1$ is for private mean estimation and $\sigma_2$ is for DP-SGD. Other hyperparameters include batch size = full, SGD optimizer with momentum = 0.9. For the linear layer, bias = False and zero initialization. We do not employ any additional regularization or learning rate schedule.

| $\varepsilon$ | Steps $T$ | $\sigma_1$ | $\sigma_2$ | learning rate |
|---|---|---|---|---|
| $\varepsilon = 1$ | 100 | 71 | 43 | 3 |
| $\varepsilon = 8$ | 200 | 14 | 9.33 | 10 |

According to Panda et al. [2024], as $\varepsilon$ increases, the total step size will also increase to achieve better performance. Therefore in Table 27, we use $T = 100$ for $\varepsilon = 1$ and $T = 200$ for $\varepsilon = 8$. In addition to Table 27, we also provide a few more observations during the hyperparameter search and include full hyperparameter search results in Table 28 and a more visually appealing representation in Fig. 8.

$T = 100$ for $\varepsilon = 8$ also achieves a compelling result that is close to 39%. When we fix $\varepsilon = 8$, as we increase $T$, the performance improves. Further increasing $T$ would continue to improve the performance. As the number of steps $T$ increases, the corresponding optimal learning rate would decrease. This is consistent with previous work [Panda et al., 2024].

TABLE 28. Hyperparameter search results for ImageNet.

| epochs<br>learning rate | 100 | 150 | 200 | 250 | 500 |
|---|---|---|---|---|---|
| 5.0 | N/A | N/A | N/A | 38.43 | 39.48 |
| 10.0 | 38.80 | 39.07 | 39.35 | 39.24 | 37.59 |
| 15.0 | N/A | N/A | N/A | 38.46 | 35.38 |
| 20.0 | N/A | N/A | N/A | 37.46 | 33.56 |
| 25.0 | 39.17 | 38.66 | 37.57 | N/A | N/A |
| 50.0 | 37.48 | 35.38 | 33.63 | N/A | N/A |
| 75.0 | 34.24 | 32.89 | N/A | N/A | N/A |

We note that Sander et al. [2023] also tried training the ViT model on ImageNet but concluded that it does not perform as well as ResNet. Our explanation for this is that ViT requires pretraining data because the architecture does not encode any natural prior, whereas CNNs naturally have a prior. The nature of convolutional filters biases CNNs to extract features with spatial locality. As we observe in Section 2, the impact of pretraining data is mostly at the initialization by giving the model a prior, so it stands to reason that the missing piece in utilizing ViT for DP training on ImageNet is learning a random prior from Shaders-21k [Baradad et al., 2022].

We also provide the hyperparameter we use for the Camelyon17 dataset for the improved linear probing result (Table 9) in Table 29.

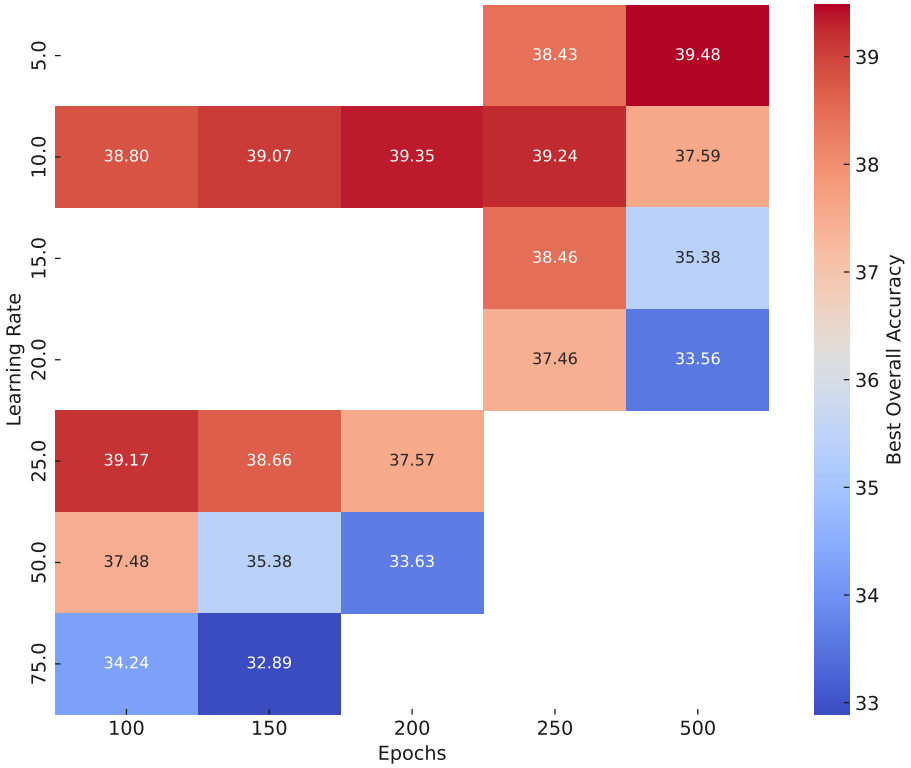


FIGURE 8. The best hyperparameters for ImageNet trend towards a larger number of epochs with a smaller learning rate.

TABLE 29. Hyperparameters for the improved linear probing on Camelyon17. $\sigma_1$ is for private mean estimation and $\sigma_2$ is for DP-SGD. Other hyperparameters include batch size = full, SGD optimizer with momentum = 0.9. For the linear layer, bias = False and zero initialization. We do not employ any additional regularization or learning rate schedule. Total number of training samples is $302,436$.

| $\varepsilon$ | Steps $T$ | $\sigma_1$ | $\sigma_2$ | learning rate |
|---|---|---|---|---|
| $\varepsilon = 10$ | 200 | 7 | 7.5 | 3 |

## APPENDIX H. TWO-STAGE TRAINING WITH PRIVATE DATA

After pretraining with synthetic data, DP-RandP first trains the classifier head (Phase II) and then tunes all hyperparameter (Phase III). Table 30 summarizes the results in the main paper and presents the comparison of our full DP-RandP, DP-RandP without Phase II and the baseline [De et al., 2022]. Note that DP-RandP without Phase III is not included in

Table 30 as training the linear layer only has diminishing returns: the non-private baseline of DP-RandP without Phase III is 74.05%, which is worse than DP-RandP without Phase II at $\varepsilon = 2$. Table 30 shows the importance of combining both Phase II and Phase III in DP-RandP.

TABLE 30. The importance of phases in DP-RandP. Evaluation of test accuracy (%) on CIFAR10.

| Phases | $\varepsilon = 1$ | $\varepsilon = 2$ | $\varepsilon = 3$ | $\varepsilon = 4$ | $\varepsilon = 6$ | $\varepsilon = 8$ |
|---|---|---|---|---|---|---|
| DP-RandP | 72.32 | 77.25 | 79.99 | 81.88 | 84.01 | 85.26 |
| DP-RandP w/o Phase II | 69.03 | 75.31 | 78.44 | 80.56 | 82.96 | 84.45 |
| Phase III only (De et al. [2022]) | 56.8 | 64.9 | 69.2 | 71.9 | 71.0 | 79.5 |

Our DP-RandP uses synthetic data for pretraining to give a warm initialization for two-stage training with private data. Table 31 shows the two-stage training with private data with random initialization. As noted in Table 16, we use $\sigma = 9.3$ for $\varepsilon = 1$ while De et al. [2022] use $\varepsilon = 10$ for the same number of 875 steps (equal to 73 epochs). This is because we only round $\sigma$ up to 0.1 and we ensure the $\sigma$ we use in Table 16 will not exceed the designed privacy bound. As we add less noise compared to De et al. [2022], the baseline result, i.e., directly updating the full parameters during training (0 epoch for Phase II), is 57.46.

TABLE 31. Test accuracy (%) of two-stage training with private data by random initialization on CIFAR10.

| Epochs for Phase II | 0 | 1 | 2 | 4 | 8 | 12 | 16 | 20 | 50 | 73 |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy (%) | 57.46 | 57.01 | 56.27 | 55.04 | 54.45 | 54.09 | 53.15 | 52.26 | 47.00 | 25.80 |
| Std. | 0.48 | 0.69 | 0.57 | 0.90 | 0.80 | 0.41 | 0.91 | 0.97 | 1.53 | 2.08 |

Table 31 shows that when the feature extractor is randomly initialized, if we train the head classifier with more steps while keeping the same total steps, the performance will decrease compared to directly fine-tuning the whole network. Our analysis of this is that, the feature extractor is not encoded with any prior and it is better to update the full network in the whole training procedure to learn more information.