Residual Connections Harm Generative Representation Learning

Xiao Zhang[*1,3] Ruoxi Jiang[*†1,2] William Gao[1] Rebecca Willet[1] Michael Maire[1]

[1]University of Chicago [2]Fudan University [3] Tencent

Abstract

We show that introducing a weighting factor to reduce the influence of identity shortcuts in residual networks significantly enhances semantic feature learning in generative representation learning frameworks, such as masked autoencoders (MAEs) and diffusion models. Our modification notably improves feature quality, raising ImageNet-1K K-Nearest Neighbor accuracy from 27.4% to 63.9% and linear probing accuracy from 67.8% to 72.7% for MAEs with a ViT-B/16 backbone, while also enhancing generation quality in diffusion models. This significant gap suggests that, while residual connection structure serves an essential role in facilitating gradient propagation, it may have a harmful side effect of reducing capacity for abstract learning by virtue of injecting an echo of shallower representations into deeper layers. We ameliorate this downside via a fixed formula for monotonically decreasing the contribution of identity connections as layer depth increases. Our design promotes the gradual development of feature abstractions, without impacting network trainability. Analyzing the representations learned by our modified residual networks, we find correlation between low effective feature rank and downstream task performance.

1. Introduction

Residual networks (ResNets) [25] define a connection structure that has achieved near-universal adoption into modern architectures for deep learning. At the time of their development, supervised learning (*e.g.,* ImageNet [16] classification) was the driving force behind the evolution of convolutional neural network (CNN) architectures. Residual networks solved a key issue: CNNs constructed of more than approximately 20 convolutional layers in sequence became difficult to train, leading to shallower networks outperforming deeper ones, unless additional techniques, such as auxiliary outputs [61] or batch normalization [33], were employed. Both ResNets, and their predecessor, highway networks [59] provide elegant solutions to this trainability problem by endowing the network architecture with alternative shortcut pathways along which to propagate gradients. Highway networks present a more general formulation that modulates these shortcut connections with learned gating functions. However, given their sufficient empirical effectiveness, the simplicity of ResNet's identity shortcuts makes them a preferred technique.

While solving the gradient propagation issue, residual connections impose a specific functional form on the network; between residual connections, each layer (or block of layers) learns to produce an update slated to be added to its own input. This incremental functional form may influence the computational procedures learned by the network [23]. Alternatives to residual and highway networks exist that do not share this functional form, but implement other kinds of skip-connection scaffolding in order to assist gradient propagation [30, 40, 71]. Thus, shortcut pathways, rather than a specific form of skip connection, are the essential ingredient to enable the training of very deep networks. Nevertheless, nearly all modern large-scale models, including those based on the transformer architecture [64] incorporate the standard residual connection.

This design choice holds, even as deep learning has shifted into an era driven by self-supervised training. The shift to self-supervision brings to the forefront new learning paradigms, including those based on contrastive [10, 11, 24, 26, 66], generative [22, 28, 36, 49, 56, 58], and autoencoding [27, 37, 42] objectives. Many systems in the generative and autoencoding paradigms rely on "encoder-decoder" architectures, often styled after the original U-Net [50], which contains additional long-range shortcuts between corresponding layers in mirrored symmetry about a central bottleneck. With representation learning as a goal, one typically desires that the middle bottleneck layer produce a feature embedding reflecting abstract semantic properties. The interaction of skip-connection scaffolding for gradient propagation with encoder-decoder architectures, self-supervised training objectives, and bottleneck representations has not been carefully reconsidered. This is a

*Both authors contributed equally to this work. This work was completed while Xiao Zhang was a graduate student at Unversity of Chicago.
†Corresponding author: roxie_jiang@fudan.edu.cn
Link to code: https://github.com/xiao7199/decayed_Identity_shortcuts

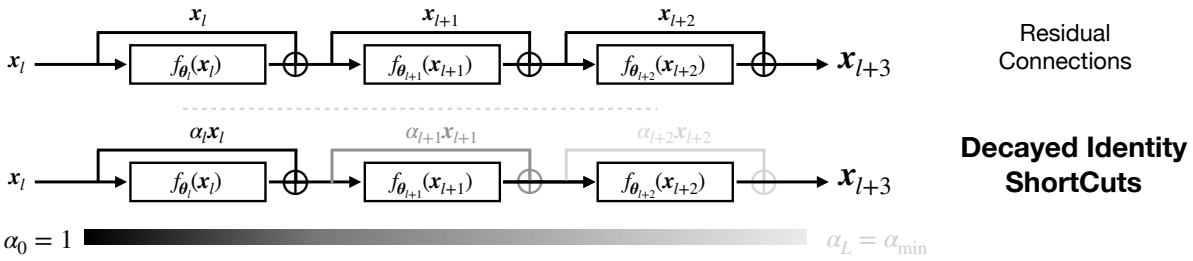


Figure 1. Our *decayed identity shortcuts* introduce a depth-dependent scaling factor to shortcuts in a residual network, thereby modulating the contribution of preceding layers and fostering greater abstraction in deeper layers. A simple schema for controlling decay factor α suffices to improve feature learning in both MAEs and diffusion models, as well as diffusion model generation quality.

worrisome oversight, especially since, even in the supervised setting with standard classification architectures, prior work suggests that unweighted identity shortcuts may be a suboptimal design decision [18, 52]. Intuitively, identity shortcuts may not be entirely appropriate for capturing high-level, semantic features as they directly inject low-level, high-frequency details of inputs into outputs, potentially compromising feature abstraction. We explore this issue within generative learning frameworks, including masked autoencoders (MAEs) [27] and diffusion models [28], leading paradigms for self-supervised image representation learning and generation. Our experiments demonstrate that identity shortcuts significantly harm semantic feature learning in comparison to an alternative we propose: gradually decay the weight of the identity shortcut over the depth of the network, thereby reducing information flow through it (Figure 1). With increasing layer depth, our approach facilitates a smooth transition from a residual to a feed-forward architecture, while maintaining sufficient connectivity to train the network effectively. Unlike prior work on learned gating [59] or reweighting [52] mechanisms for residual connections, our method is a forced decay scheme governed by a single hyperparameter.

A parallel motivation for our design stems from Huh et al. [32], who show that features from residual blocks have higher rank than those produced by comparative feed-forward blocks. The smooth transition between residual and feed-forward behavior induced by our decay scheme regularizes deeper features toward exhibiting low-rank characteristics. Section 6 experimentally explores the correlation between our decayed identity shortcuts and low-rank feature representations. Figure 2 previews the corresponding improvements to feature learning. Our contributions are:

- We introduce decayed identity shortcuts, a simple architectural mechanism which enhances feature abstraction in masked autoencoders and diffusion models.
- Our design within an MAE yields a substantial performance boost on ImageNet-1K [16]: achieving a linear probing accuracy of 72.7% (up from a baseline of 67.8%) and a K-Nearest Neighbor accuracy of 63.9% (an improvement from the baseline of 27.4%).
- In diffusion models, our design improves both feature learning and generation quality.
- Smaller models with decayed identity shortcuts outperform larger ones using standard residual connections.

2. Related Work

Self-supervised representation learning. Recent advancements [1, 38, 48, 49, 54, 62] in deep learning follow a common scaling law, in which a model's performance consistently improves with its capacity and the size of the training data. This effect can be observed in large language models (LLMs), which are trained on vast amounts of internet text, enabling them to perform some tasks at a human level [41] and exhibit remarkable zero-shot capabilities [39]. These models are trained using next-token-prediction, allowing them to be trained without labeled data. In contrast, the progress of this scaling law in computer vision has largely depended on annotated data. For instance, the Segment Anything [38] leverages 1 billion human-annotated masks, and state-of-the-art image generators [48] require training on huge datasets of text-image pairs [53]. However, the vast volume of unlabeled visual data and desire for continued scaling motivates a transition to self-supervised learning.

At present, two families of approaches to self-supervised visual representation learning appear particularly promising. **Contrastive representation learning** [10, 11, 24, 26, 66] achieves state-of-the-art performance in most downstream classification tasks by training discriminative models to maximize mutual information between differently augmented views of images. However, these approaches rely on extensive and intricate data augmentation pipelines, necessitating domain expertise for adaptation to new domains. **Generative representation learning**, via masked image modeling [6, 14, 27], which trains to reconstruct occluded pixels, or via diffusion denoising [28, 56, 57], which trains to reverse a process that mixes images with Gaussian noise, relying less on forming discriminative augmentations, learns to extract representations inherently along the

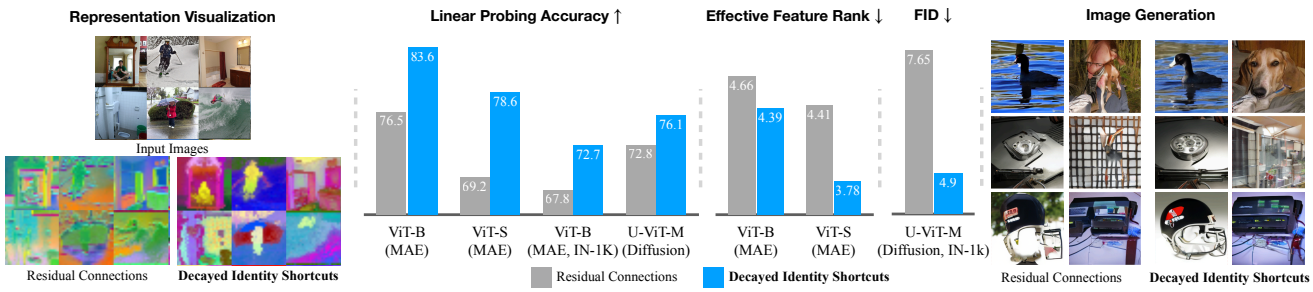


Figure 2. We design *decayed identity shortcuts* (Figure 1), a variant of residual connections, to facilitate self-supervised representation learning in generative model. Compared to standard residual connections, our approach yields superior abstract semantic features (*left*, visualized using Zhang et al. [69]'s approach), whose leading components pop out object instances and classes. Quantitative evaluation shows our architecture encourages lower feature rank and learns better feature representation for both MAE and diffusion models (*middle*), along with enhanced generation quality for diffusion models (*right*). These improvements require no additional learnable parameters.

generative process. Some hybrid approaches [31, 42, 70] combine both families. Despite advancements, neither has demonstrated the same scalability [55] as seen in LLMs. This challenge is additional motivation for reconsidering the foundations of self-supervised network architectures.

Residual and skip-connection architectures. Highway networks [23] first propose an additive skip connection structure to provide a scaffolding for gradient propagation when training very deep (*e.g.,* 100 layer) networks. Motivated by the gating mechanisms within LSTMs [29], this solution uses learned gating functions to weight each combination of identity and layer output branches. Residual networks [25] are a simplification that removes these learned coefficients. DenseNet [30] and FractalNet [40] demonstrate that access to gradient paths of multiple lengths is the core requirement of training scaffolding, by introducing skip-connection structures with other functional forms. DenseNet utilizes feature concatenation instead of addition, while FractalNet imposes a recursive tree-like architecture combining subnetworks of multiple depths.

Zhu et al. [71] explore variants of ResNets and DenseNets with fewer points of combination between different internal paths, demonstrating that a sparser scaffolding structure may be more robust as network depth increases to thousands of layers. Savarese and Figueiredo [52] add a scalar gating functional to the layer output in residual networks, yielding a hybrid design between residual and highway networks; learning this scalar gating provides a consistent benefit to classification accuracy. Fischer et al. [18] develop a weighting scheme for residual connections based upon a sensitivity analysis of signal propagation within a ResNet. To date, none of these potential improvements has seen broad adoption.

Low rank bias in neural networks. Over-parameterized neural networks exhibit surprising generalization capabilities, a finding seemingly in contradiction with classical machine learning theory [45]. This phenomenon implies the existence of some form of implicit regularization that prevents the model from overfitting. From the perspective of model parameterizations, Arora et al. [2] suggest that linear models with more layers tend to converge to minimal norm solutions. In the context of CNNs, Huh et al. [32] demonstrate that stacking more feed-forward layers compels the model to seek lower rank solutions, and Jing et al. [34] reinforce this finding by adding more layers to an autoencoder's bottleneck, thereby creating a representation bottleneck. In vision transformers, Geshkovski et al. [20] examine the connection between attention blocks and mean-shift clustering [15], showing that repeated attention operations result in low-rank outputs. Moreover, Dong et al. [17] reveal that eliminating the shortcut connection from residual attention blocks causes features to degenerate to rank 1 structures doubly exponentially. From a different perspective, recent work [8, 46, 47] shows training algorithms implicitly induce low-rank behavior in neural networks. Radhakrishnan et al. [47] study the dimensionality reduction behavior of a recursive feature machine [46] and effectively verify performance on low-rank matrix recovery.

3. Method

In this section, we present the methodology for promoting a low-rank inductive bias using our proposed decay schema and discuss additional implementation strategies designed to stabilize the training process.

3.1. Decayed Identity Shortcuts

Feed-forward layers. Consider a neural network of L layers. For each layer l parameterized with θ_l, the operation of a feed-forward neural network can be described as:

$$\boldsymbol{x}_{l+1} = f_{\theta_l}(\boldsymbol{x}_l), \tag{1}$$

where $\boldsymbol{x}_l \in \mathbb{R}^d$ represents the output from the preceding layer, and f_{θ_l} denotes the network block applied at the cur-

rent layer. Although it is widely known that pure feedforward architectures are susceptible to vanishing gradients when building deeper models, Huh et al. [32] demonstrates that feed-forward modules offer implicit structural regularization, enabling deep models to generate abstract representations at bottlenecks.

Residual connections. To address the optimization problem of vanishing gradients in deeper neural networks, ResNets [25] construct each layer as a residual function, resulting in a modification to Eq. 1:

$$\boldsymbol{x}_{l+1} = \boldsymbol{x}_l + f_{\theta_l}(\boldsymbol{x}_l). \tag{2}$$

This design builds shortcuts from input to output, allowing gradient magnitude to be preserved regardless of the depth of the model. However, a consequence of this design is that the output stays close to the input in practice [23], defeating the need to construct complex transformations over depth. The same phenomenon is also observed in highway networks [59], which adopt learnable gates $H_\phi(\boldsymbol{x}) \in [0,1]^d$ in both the residual and skip branches: $\boldsymbol{x}_{l+1} = H_\phi(\boldsymbol{x}_l) \cdot \boldsymbol{x}_l + (1 - H_\phi(\boldsymbol{x}_l)) \cdot f_{\theta_l}(\boldsymbol{x}_l)$. Although this flexible design allows the model to build the abstraction level over depths, similar to feedforward networks, Srivastava et al. [60] finds $H_\phi \approx 1$ for most units, suggesting the model prefers copying the input.

Decayed identity shortcuts for unsupervised representation learning. Setting aside the optimization benefits brought by residual connections, we rethink the role of the residual connections from the viewpoint of representation learning. Abstraction can be viewed as invariance to local changes of input and is crucial to the disentanglement of the feature space [9]. Prior work suggests that a shortcut path of residual connections tends to preserve high-frequency fine-grained input information [23], resulting in decreased feature abstraction. We hypothesize that this lack of abstraction harms the capability of the model to learn meaningful low-level features and that ensuring an abstract structure in the deeper layers of the neural network will help improve representation learning, especially for unsupervised tasks that often use indirect proxy objectives, such as pixel-wise reconstruction loss. Motivated by this hypothesis, we propose to downweight the contribution from the shortcut path:

$$\boldsymbol{x}_{l+1} = \alpha_l \boldsymbol{x}_l + f_{\theta_l}(\boldsymbol{x}_l), \tag{3}$$

where $\alpha_l \in [0,1]$ is a rescaling factor to the residual path, controlling the information flow through the skip connection. Fully expanding this relation for a network with L layers indexed from 1 to L, we have that:

$$\boldsymbol{x}_{L+1} = \left(\prod_{l=1}^{L} \alpha_l\right) \boldsymbol{x}_0 + \sum_{l=1}^{L-1} \left(\prod_{i=l+1}^{L} \alpha_i\right) f_{\theta_l}(\boldsymbol{x}_l) + f_{\theta_L}(\boldsymbol{x}_L). \tag{4}$$

We see that the contribution of the input $\boldsymbol{x}_0$ is scaled by each $\alpha_l \leq 1$ while each subsequent network block output $f_{\theta_l}(\boldsymbol{x}_l)$ omits scaling factors up to α_l. Hence, the contribution of early features of the network is especially downweighted, preventing the network from passing fine-grained detailed information to the bottleneck X_{L+1}. During our experiments, we find the effective decay factor of the final layer, $\alpha_L^{\text{eff}} = \prod_{l=1}^{L} \alpha_l$, plays a critical role in deciding the optimal decay rate when varying the network depth L.

Decay schema. Instead of specifying α_l as a constant across all layers, we choose α_l to be a function parameterized by the layer index l, where the contribution from the shortcut path is monotonically decreasing when l increases:

$$\alpha_l = 1 - \delta_\alpha l, \tag{5}$$

where $\delta_\alpha := \frac{(1-\alpha_{\min})}{L}$, $\alpha_L \equiv \alpha_{\min}$ is a minimum scaling factor applied at the final layer L. Our formulation brings two primary benefits. First, α_l, as a linear interpolation between 0 and 1, acts as a smooth transition between residual connections and feedforward layers, bringing us the optimization benefits from residual connections, while simultaneously encouraging the deeper layers to learn more abstract representations. Second, similar to the naive formulation, our method only introduces one extra hyperparameter $\alpha_{\min}$, which is not data-dependent and does not need to be learned.

3.2. Implementation Strategy

Skip connections for autoencoders. Since our method progressively decays the residual connections over network depth, it encourages the most abstract features to be learned by later layers. However, learning an abstract bottleneck is detrimental to the training objectives that aim for pixel-wise reconstruction, as they necessitate the preservation of detailed information. To address this, we incorporate standard skip connections between the encoder and decoder, enabling the encoder to directly pass information from shallow layers to the decoder while learning increasingly abstract representations in the deeper encoder layers.

Stabilizing training with residual zero initialization. The model exhibits rapid feature norm growth at the beginning of training for $\alpha_{\min} \leq 0.7$. We suspect that the model learns to amplify the output feature norm of $f_{\theta_l}(\boldsymbol{x})$ to counteract the significant decay applied to the residual connection. This growth leads to training instability and negatively impacts training convergence. To address this issue, we follow the implementation of previous works [28] and initialize the weights of the final output layer in each f_{θ_l} to zero instead of using the original Xavier uniform initialization [21]. This approach enhances training stability by controlling the growth of feature norm, especially with smaller $\alpha_{\min}$.

Method	FT	LP	KNN
Contrastive representation learning			
MoCo-v3[12]	83.2	76.7	66.6
DINO[10]	83.3	**78.2**	**76.1**
Con MIM [68]	**83.7**	39.3	-
Generative representation learning			
Data2Vec[4]	84.2	68.0	33.2
I-JEPA[3]	-	**72.9**	-
CAE[14]	83.8	70.4	51.4
ADDP(VIT-L) [63]	**85.9**	23.8	-
Latent MIM[65]	83.0	72.0	50.1
MAE[27]	83.6	67.8	27.4
MAE ($\alpha_{\min} = 0.6$)	82.9	**72.7**	**63.9**

Table 1. **Benchmark of representations on the ImageNet-1K with ViT-B.** Evaluate learned features using standard evaluation protocols: linear probing (LP), fine-tuning (FT) and K-Nearest Neighbor (KNN). With only a simple architectural modification to MAE [27] and trained purely with pixel-wise reconstruction loss, we achieve 72.7% LP accuracy and 63.9% KNN accuracy, significantly narrowing down the gap between generative and contrastive representation learning frameworks

Method	Train Iters	UNet	FID	IS
SiT-XL/2 (Baseline)	200k	False	25.2	-
SiT-XL/2 ($\alpha_{\min} = 1.0$)	200k	True	22.7	57.3
SiT-XL/2 ($\alpha_{\min} = 0.8$)	200k	True	**14.2**	**79.7**
SiT-XL/2 (Baseline)	400k	False	17.2	-
SiT-XL/2 ($\alpha_{\min} = 1.0$)	400k	True	16.5	74.8
SiT-XL/2 ($\alpha_{\min} = 0.8$)	400k	True	**11.8**	**91.8**

Table 2. **Class-conditional ImageNet-1k 256x256 Generation with SiT-XL/2 [44].** Our method significantly enhances the generation performance and the convergence speed. Comparison between ours ($\alpha_{\min} = 0.8$) and UNet baseline ($\alpha_{\min} = 1.0$) suggests the improvements are primarily from our decayed identity shortcuts rather than the encoder-decoder skip connections.

4. Experiments on Masked Autoencoders

For masked autoencoders (MAEs) [27], we replace the residual connections in the encoder's MLP and attention blocks with decayed identity shortcuts. The MAE operates by accepting images with a random subset of pixels masked out and learning to recover the discarded pixels. Since the original MAE has twice the number of encoder layers as decoder layers, we build encoder-decoder skip connections by injecting output from every other encoder layer into the corresponding decoder layer. To match spatial dimensions, injected encoder features are combined with learnable masked tokens before channel-wise concatenation. The implementation details for the training and evaluation are shown in Section A. He et al. [27] show the desired representations appear at the end of encoder; we therefore apply our decaying schema only to the encoder.

4.1. Representation Learning on ImageNet-1k

We follow the default hyperparameters from MAE [27] to pretrain ImageNet-1K train split [16] and use the standard protocol to evaluate the learned representation with end-to-end finetuning (FT), linear probing (LP) and K-Nearest Neighbour (KNN, K = 20), for image classification task. Please see the appendix for detailed experimental setups.

We report the results in Table 1. In the top half of the table, we present methods that employ a contrastive loss. Although these methods produce the best probing accuracies, their success depends on a carefully designed data augmentation process, which may need to be tuned for each different data distribution. In the bottom half, we show several methods based on generative architecture. Our method simply extends MAE by constructing an implicit feature bottleneck and shows significant improvements over the MAE baselines for both linear probing (72.7% *vs.* 67.3%) and K-Nearest Neighbour (63.9% *vs.* 27.4%). outperforming Data2Vec, Latent MIM and CAE and giving a probing accuracy competitive with I-JEPA, without needing to use explicit feature alignment.

End-to-end fine-tuning (FT), unlike linear probing which only trains a single linear layer, updates the entire network for image classification. Since the features can shift significantly from their pre-training state during end-to-end updating, we argue that this may not accurately reflect the quality of the learned representations. For example, DINO demonstrates superior performance in various downstream vision tasks compared to MAE, but its fine-tuning performance is worse than MAE. Similarly, ConMIM and ADDP exhibit poor linear probing performance, suggesting lower-quality representations, yet their fine-tuning performance surpasses that of contrastive learning methods. Nevertheless, we still provide the fine-tuning results for reference.

4.2. Ablation Studies on ImageNet-100

We conduct ablations on several properties of our framework on ImageNet-100. A summary of results can be found in Tables 3 and 4.

Decay rate $\alpha_{\min}$. The only parameter of our framework is $\alpha_{\min}$, the minimum scaling factor applied to the final layer. In Table 3,we show linear probing scores for varying values of $\alpha_{\min}$. We observe that $\alpha_{\min} \in [0.6, 0.7]$ works well for most cases. If $\alpha_{\min}$ is too small, for example, $\alpha_{\min} \leq 0.4$, we observe that the training becomes unstable.

Architecture size. In Table 3, we also run experiments over multiple architecture choices. We notice encoder depth L rather than feature dimension that influences the optimal choices of $\alpha_{\min}$ and deeper models require larger $\alpha_{\min}$. We attribute this phenomenon to the cumulative decaying effect of the final layer, quantified as $\alpha_L^{\text{eff}} = \prod_{l=1}^{L} \alpha_l$. Heavy decaying would harm the optimization and selecting $\alpha_{\min}$ such that $\alpha_L^{\text{eff}} \in$ [1e-3, 1e-2) yields the best performance.

Feat. Dim.	Enc. Depth (L)	$\alpha_{\min}$					α_L^{eff}			
		0.6	0.7	0.8	0.9	1.0 (Baseline)	(0, 1e-3)	[1e-3, 1e-2)	[1e-2, 1e-1)	[1e-1, 1]]
384	12	**78.5**	78.1	75.2	73.5	69.2	-	**78.5**	78.1	75.2
768	12	**83.6**	81.8	79.8	79.2	76.5	-	**83.6**	81.8	79.8
1024	12	**83.2**	82.5	82.1	79.3	78.0	-	**83.2**	82.5	82.1
768	18	83.5	**85.0**	84.4	81.8	79.2	78.5	**85.0**	84.4	81.8
1024	24	84.3	**86.0**	84.5	84.3	81.4	82.4	**86.0**	-	84.3

Table 3. **Linear Probing accuracy of MAE on ImageNet-100 varying $\alpha_{\min}$ and architecture.** We show our method consistently improves the performance across all configurations. We notice the encoder depth rather than feature dimension influences the optimal $\alpha_{\min}$. We attribute this behavior to the scaling effect of the input data to encoder's final layer, quantified as $\alpha_L^{\text{eff}} = \prod_{l=1}^{L} \alpha_l$. Deeper model requires larger $\alpha_{\min}$ to maintain a consistent cumulative decay effect and we find setting $\alpha_{\min}$ such that $\alpha_L^{\text{eff}} \in$ [1e-3, 1e-2) works the best. With our strategy, smaller model (768 feat. dim + 12 layer) outperforms bigger one (1024 feat. dim + 24 layer) with standard residual connections.

$\alpha_{\min}$	α_l **scheduler**	**UNet**	**LP**
0.6	Linear	Yes	**83.6**
0.6	Linear	No	61.5
0.6	Cosine	Yes	82.8
0.7	Linear	Yes	81.8
0.7	Cosine	Yes	82.9
-	Learnable α_l	Yes	79.5

(a) **Effect of Skip Connections and α_l scheduler**. Skip connection is critical for performance. The choice of schedulers has less impact and learnable α_l is worse than pre-fixed α_l.

Configurations	**Decay Block**	$\alpha_{\min}$	**LP**
$\boldsymbol{x}_{l+1} = \boldsymbol{x}_l + f_{\theta_l}(\boldsymbol{x}_l)$	-	–	76.5
$\boldsymbol{x}_{l+1} = \boldsymbol{x}_l + \sqrt{0.5} f_{\theta_l}(\boldsymbol{x}_l)$	MLP & Atten.	–	76.9
$\boldsymbol{x}_{l+1} = \sqrt{0.5}\,(\boldsymbol{x}_l + f_{\theta_l}(\boldsymbol{x}_l))$	MLP & Atten.	–	82.6
$\boldsymbol{x}_{l+1} = \alpha_l \boldsymbol{x}_l + f_\theta(\boldsymbol{x}_l)$	Atten.	0.6	79.3
$\boldsymbol{x}_{l+1} = \alpha_l \boldsymbol{x}_l + f_\theta(\boldsymbol{x}_l)$	MLP	0.6	80.6
$\boldsymbol{x}_{l+1} = \alpha_l \boldsymbol{x}_l + f_\theta(\boldsymbol{x}_l)$	MLP & Atten.	0.6	**83.6**

(b) **Other Decay Schemas**. We conduct ablations using a variety of scalings of the residual connection and observe that our design produces the best results.

Table 4. **Ablation experiments of MAE using ViT-B/16 in ImageNet-100.** We ablate decay scheduler and architectural design with linear probing (LP) accuracy.

Skip connections. Another critical design choice in our network is to include skip connections that are not in the original MAE. As discussed in Section 3.2, if the MAE does not use skip connections, the bottleneck layer must preserve all information to reconstruct the input image accurately. This is opposed to learn abstract representations at bottleneck. These contrary effects significantly degrade the representation learned by the model, leading to a 22.1% drop in the linear probing score, as we report in Table 4a.

α_l Scheduler. We use linear scheduler $\alpha_l = 1 - \frac{(1-\alpha_{\min})}{L} l$ as a default choice. In table 4a, we also experiment with a cosine scheduler but find it leads to worse performance for $\alpha_{\min} = 0.6, 0.7$. Besides using a prefixed α_l scheduler, we also experiment with a learnable α_l, which resembles the setup of highway network [59]. We show the learnable α_l at each layer in Table6, appendix. From the table, we don't find consistent patterns over network depth and the performance is worse than our predefined α_l scheduler.

Different decay schema. We also explore decay schema, with results summarized in Table 4b: (1) Scaling both branches of the residual blocks simultaneously by applying a constant factor, $\alpha = \sqrt{0.5}$, to both $\boldsymbol{x}$ and $f_{\theta_l}(\boldsymbol{x})$. (2) Scaling only f_{θ_l} using the same constant factor, $\alpha = \sqrt{0.5}$. (3) Applying our proposed schema exclusively to either the attention or MLP branch.

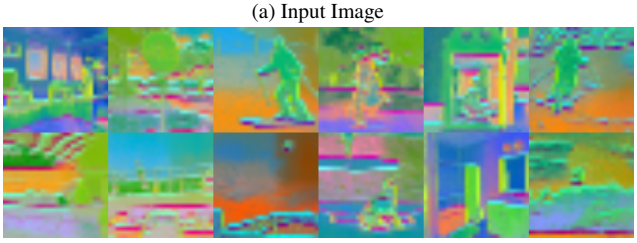
Among these, (2) shows no significant improvement over the baseline, while (1) yields some improvement but still underperforms compared to our approach. By analyzing (1) and (2), we demonstrate that the representation gains are due to down-weighting the skip connection branch. Notably, recent diffusion models [35, 57] have employed (1) in their smaller convolutional neural network but don't provide systemic analysis. However, applying decay only to the MLP or attention branch reduces the overall decaying effect across the network, resulting in lower performance compared to our schema, which achieves the best performance among the tested designs.

4.3. Embedding Analysis

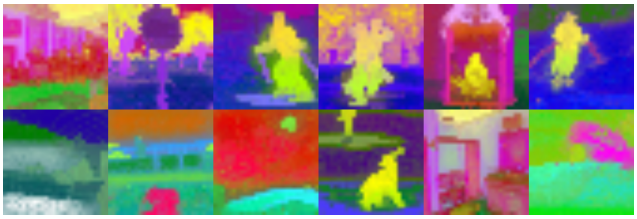
We qualitatively evaluate the feature learning in Figure 3 and we adopt the pixel-wise embedding approaches proposed by Zhang et al. [69] to group the representations from the last layer of the encoder into a lower dimensional space. We use their default hyperparameters to cluster representations across COCO validation set and render the top 3 eigenvectors as RGB channels of images. From the visualization, ours learns abstract representation and the object from the same categories have similar color, indicating a global consistent semantic grouping. The baseline MAE, on the other hand, doesn't show clearly global semantic patterns.

We also benchmark the clustering quantitatively, following the postprocessing protocol [69] to produced un-

(a) Input Image



(b) MAE (4.1 mIoU)



(c) MAE ($\alpha_{\min}$ = 0.6) (10.4 mIoU)

Figure 3. **Visualize learned representations using Zhang et al. [69] without cherry-picking.** We project the learned representations onto a 3-channel feature map, visualized as RGB images. Our method learns more abstract and semantically consistent representations. This visual comparison is further supported by benchmarking on unsupervised semantic segmentation tasks, where our approach achieves better results (10.4 mIoU) compared to the baseline MAE (4.1 mIoU) on COCO validation set.

supervised semantic segmentation and report the results as the mean intersection of union (mIoU). Ours (10.41 mIoU) achieve 6.31 mIoU improvement over baseline (4.10 mIoU), which support the visual comparison.

5. Experiments on Diffusion Models

Diffusion models. We use U-ViT [5], a ViT-based diffusion model with skip connections between the encoder and decoder, and SiT-XL/2 [44] as the baseline for our diffusion model experiments. To adapt SiT-XL/2 to our design, we add skip connection from the first half of the layers to the corresponding second half of the layers. Unlike in MAE where we only apply decayed connections within the encoder, in diffusion models, we decay the skip connections till the last layer of the decoder, following the recent studies [7, 67] suggesting that diffusion models learn the best semantic representations near the decoder's latter stages. While this design might be suboptimal, as the smallest decay factor may not align with the layers holding the best semantic representations, we demonstrate in practice that this simple approach effectively enhances both the learned representations and the quality of generated outputs.

Experimental details. We utilize the default scheduler and sampler from U-ViT [5] and SiT-XL/2 [44] respectively. We train class-conditional diffusion model on ImageNet-100 and ImageNet-1k and we additionally train unconditional diffusion models on CIFAR-100 and ImageNet-100 without using image class labels to validate our design across tasks. For ImageNet-100 and ImageNet-1k, we follow the setup in latent diffusion models [49] by running the model in the latent space of a pretrained VAE, which maps input images with 256x256x3 to a 32x32x4 sampled latent space. In ablation experiments with U-ViT, We use U-ViT-Mid for ImageNet-100 and ImageNet-1K, and U-ViT-small for CIFAR-100. For model design and training details, please refer to Bao et al. [5].

We evaluate the learned representations with linear probing and we train a linear classifier over the frozen representations. We report the results as the best configurations, including the choices of layer index and noise level, that yields the best performance.

Results. We report our primary ImageNet-1k 256x256 class-conditional image generation performance with SiT-XL/2 in Table 2. Compared to baseline, we demonstrate significant improvement in FID: (11.8 *v.s.* 17.2). Further, comparing ours with SiT-XL/2 ($\alpha_{\min}$ = 1.0) (11.8 *v.s.* 16.5) further validates that our improvements are primarily from our decay scheme rather than the encoder-deocder skip connection. Our simple design also shows significant convergence speed up where ours at 200K steps are substantially better than baseline at 400 steps (14.2 *v.s.* 17.2).

Besides, we also provide ablation with U-ViT models across multiple dataset and experimental setup. We present our results in Table 5, where we demonstrate that replacing residual connections with our proposed decayed identity shortcuts consistently enhances representation quality and image generation across both datasets and tasks (conditional and unconditional generation). Notably, this improvement is achieved without introducing any additional learnable parameters. We provide the visualization of generated images in the appendix for qualitative comparisons (Figure 8).

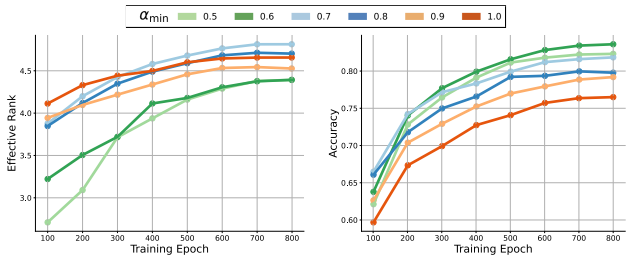
6. Discussion on Feature Rank

In this section, we try to provide insights for the key question: How and why do residual connections impact the abstraction level of the deeper layers in a neural network? We delve deeper into how our design reinforces the low-rank bias of neural networks and try to connect our method to ideas in existing works [32]. To this end, we visualize the training dynamics of our method and analysis the feature rank of our approach to provide a holistic analysis.

Low-rank simplicity bias. Huh et al. [32] investigate the low-rank simplicity bias in deeper feed-forward neural net-

Dataset \ $\alpha_{\min}$	Model	Linear Probing (Acc.)↑				Generation quality (FID)↓			
		1.0	0.8	0.7	0.6	1.0	0.8	0.7	0.6
CIFAR-100 (Uncon.)	U-ViT-Small	62.47	63.58	**66.86**	64.63	14.34	11.65	**8.99**	11.71
ImageNet-100 (Uncond.)	U-ViT-Mid	72.8	74.5	**76.1**	75.8	44.40	**40.96**	41.17	43.51
ImageNet-100 (Class Cond.)	U-ViT-Mid	-	-	-	-	6.93	5.75	5.11	**4.98**
ImageNet-1K (Class Cond.)	U-ViT-Mid	-	-	-	-	7.65	6.23	5.34	**4.90**

Table 5. **Ablation on image generation using U-ViT models.** In diffusion models, we demonstrate that our proposed decayed identity shortcut (with $\alpha_{\min} < 1.0$) enhances probing accuracy and generation quality across various configurations in both classes conditional and unconditional setups.



(a) Effective Rank of MAE over training epochs.

(b) Linear probing accuracy of MAE over training epochs.

Figure 4. For MAE pretrained on ImageNet-100, we present visualizations of (a) the training dynamics of the effective rank for different values of $\alpha_{\min}$, and (b) the linear probing accuracy for various $\alpha_{\min}$, demonstrating that a lower effective feature rank could potentially be associated with better performance.

works, which drives neural networks to find low-rank solutions. At the same time, they make an empirical observation that deeper residual networks do not show a similar rank contracting behavior.

Effective rank. For analysis purpose, Huh et al. [32] quantify the rank of the learned representation using the *effective rank*, which is a continuous measure. For a matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, the effective rank $\rho(\boldsymbol{A})$ is defined as the Shannon entropy of the normalized singular values [51]: $\rho(\boldsymbol{A}) = -\sum_i^{\min(n,m)} \bar{\sigma}_i \log \bar{\sigma}_i$ where $\bar{\sigma}_i = \sigma_i / \sum_j \sigma_j$ denotes the i^{th} normalized singular value. Intuitively, $\rho(\boldsymbol{A})$ is small when a few singular values dominate and large when singular values are evenly spread, hence giving a good continuous approximation for matrix rank. In the following subsections, we use the singular values from the covariance matrix $\boldsymbol{A}_\theta$ of the last-layer features to compute $\rho(\boldsymbol{A})$, where $\boldsymbol{A}_\theta(i, j)$ denotes the covariance of the learned class tokens for the i^{th} and j^{th} samples.

Inspired by Huh et al. [32], *we conjecture that the feature learning improvement of our method can potentially be attributed to the decayed identity shortcuts encouraging low-rank features at the bottleneck.* In Figures 4a and 4b, we measure the training dynamics of the models (feature dimension 768 and encoder depth 12) in terms of accuracy and the effective rank, for different values of $\alpha_{\min}$. In early epochs, models with lower $\alpha_{\min}$ tend to exhibit both lower effective rank and higher probing accuracy, supporting our hypothesis. As training progresses, the correlations between $\alpha_{\min}$ and effective rank become less precise. We suspect this is due to the model's effort to compensate for the decay factors with a large value. And we can still see that lower $\alpha_{\min}$ =[0.5-0.6] results in lower feature rank and better probing accuracy compared to $\alpha_{\min}$ = [0.7-1.0]. We provide further analysis in Appendix B.3.

Compatibility with contrastive learning frameworks. Despite substantial improvement of applying our decayed identity shortcuts in generative models, we note that our approach does not easily extend to contrastive learning frameworks, where the low rank inductive bias conflicts with the training objectives, e.g., MoCov3 [13] include a universal repulsive term in the denominator to increase the feature rank. Rankme [19] confirms this by showing contrastive learning models prefer higher feature ranks.

7. Conclusion

Huh et al. [32] raise a key insight in their work – that how a neural network is parameterized matters for fitting the data – and investigate the inductive low-rank bias of stacking more linear layers in a network. In this work, we observe that the ubiquitous residual network [25] may not be the ideal network parametrization for representation learning and propose a modification of the shortcut path in residual blocks that significantly improves unsupervised representation learning. We explore the connection between our reparameterization of the residual connection and the effective rank of the learned features, finding a potential correlation between good representations and low-rank representations.

Our work calls into question a fundamental design choice of neural networks that has been used in many modern architectures. By rethinking this choice, the door is open for further reparametrizations and improvements to unsupervised representation learning. The results we show provide a prompt for more extensive investigations into the connection between low effective rank and high-quality abstract representations.

8. Acknowledgments

We gratefully acknowledge the support of AFOSR FA9550-18-1-0166, NSF DMS-2023109, DOE DE-SC0022232, the NSF-Simons AI-Institute for the Sky (SkAI) via grants NSF AST-2421845 and Simons Foundation MPS-AI-00010513, and the Margot and Tom Pritzker Science Foundation.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv:2303.08774*, 2023. 2

[2] Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In *NeurIPS*, 2019. 3

[3] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *CVPR*, 2023. 5

[4] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *ICML*, 2022. 5

[5] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A ViT backbone for diffusion models. In *CVPR*, 2023. 7

[6] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *ICLR*, 2022. 2

[7] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *ICLR*, 2022. 7

[8] Daniel Beaglehole, Adityanarayanan Radhakrishnan, Parthe Pandit, and Mikhail Belkin. Mechanism of feature learning in convolutional neural networks. *arXiv:2309.00570*, 2023. 3

[9] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *TPAMI*, 2013. 4

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 1, 2, 5

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1, 2

[12] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 5

[13] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *ICCV*, 2021. 8

[14] Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. Context autoencoder for self-supervised representation learning. *IJCV*, 2024. 2, 5

[15] Yizong Cheng. Mean shift, mode seeking, and clustering. *TPAMI*, 1995. 3

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 2, 5

[17] Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *ICML*, 2021. 3

[18] Kirsten Fischer, David Dahmen, and Moritz Helias. Optimal signal propagation in ResNets through residual scaling. *arXiv:2305.07715*, 2023. 2, 3

[19] Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International conference on machine learning*, pages 10929–10974. PMLR, 2023. 8

[20] Borjan Geshkovski, Cyril Letrouit, Yury Polyanskiy, and Philippe Rigollet. The emergence of clusters in self-attention dynamics. In *NeurIPS*, 2024. 3

[21] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *ICAIS*, 2010. 4

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1

[23] Klaus Greff, Rupesh K Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation. In *ICLR*, 2017. 1, 3, 4

[24] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020. 1, 2

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 3, 4, 8

[26] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1, 2

[27] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1, 2, 5, 12

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2, 4

[29] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 1997. 3

[30] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 1, 3

[31] Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. Contrastive masked autoencoders are stronger vision learners. *TPAMI*, 2023. 3

[32] Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks. *arXiv:2103.10427*, 2021. 2, 3, 4, 7, 8

[33] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 1

[34] Li Jing, Jure Zbontar, and Yann LeCun. Implicit rank-minimizing autoencoder. In *NeurIPS*, 2020. 3

[35] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018. 6

[36] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *TPAMI*, 2021. 1

[37] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013. 1

[38] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 2

[39] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *NeurIPS*, 2022. 2

[40] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. FractalNet: Ultra-deep neural networks without residuals. In *ICLR*, 2017. 1, 3

[41] Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Xiangji Huang. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. *arXiv:2305.18486*, 2023. 2

[42] Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. MAGE: Masked generative encoder to unify representation learning and image synthesis. In *CVPR*, 2023. 1, 3

[43] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 12

[44] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024. 5, 7

[45] Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. In *ICLR*, 2019. 3

[46] Adityanarayanan Radhakrishnan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin. Mechanism of feature learning in deep fully connected networks and kernel machines that recursively learn features. *arXiv:2212.13881*, 2022. 3

[47] Adityanarayanan Radhakrishnan, Mikhail Belkin, and Dmitriy Drusvyatskiy. Linear recursive feature machines provably recover low-rank matrices. *arXiv:2401.04553*, 2024. 3

[48] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 2

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 2, 7

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1, 12

[51] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *ESPC*, 2007. 8

[52] Pedro Savarese and Daniel Figueiredo. Residual gates: A simple mechanism for improved network optimization. In *ICLR*, 2017. 2, 3

[53] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. LAION-5B: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 2

[54] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. In *ACL*, 2020. 2

[55] Mannat Singh, Quentin Duval, Kalyan Vasudev Alwala, Haoqi Fan, Vaibhav Aggarwal, Aaron Adcock, Armand Joulin, Piotr Dollár, Christoph Feichtenhofer, Ross Girshick, et al. The effectiveness of MAE pre-pretraining for billion-scale pretraining. *arXiv:2303.13496*, 2023. 3

[56] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1, 2

[57] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 2, 6

[58] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023. 1

[59] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv:1505.00387*, 2015. 1, 2, 4, 6

[60] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In *NeurIPS*, 2015. 4

[61] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1

[62] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: A family of highly capable multimodal models. *arXiv:2312.11805*, 2023. 2

[63] Changyao Tian, Chenxin Tao, Jifeng Dai, Hao Li, Ziheng Li, Lewei Lu, Xiaogang Wang, Hongsheng Li, Gao Huang, and Xizhou Zhu. ADDP: Learning general representations for image recognition and generation with alternating denoising diffusion process. In *ICLR*, 2024. 5

[64] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1

[65] Yibing Wei, Abhinav Gupta, and Pedro Morgado. Towards latent masked image modeling for self-supervised visual representation learning. *arXiv preprint arXiv:2407.15837*, 2024. 5

[66] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 1, 2

[67] Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *ICCV*, 2023. 7

[68] Kun Yi, Yixiao Ge, Xiaotong Li, Shusheng Yang, Dian Li, Jianping Wu, Ying Shan, and Xiaohu Qie. Masked image modeling with denoising contrast. In *ICLR*, 2023. 5

[69] Xiao Zhang, David Yunis, and Michael Maire. Deciphering 'what' and 'where' visual pathways from spectral clustering of layer-distributed neural representations. In *CVPR*, 2024. 3, 6, 7

[70] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pre-training with online tokenizer. In *ICLR*, 2022. 3

[71] Ligeng Zhu, Ruizhi Deng, Michael Maire, Zhiwei Deng, Greg Mori, and Ping Tan. Sparsely aggregated convolutional networks. In *ECCV*, 2018. 1, 3