Are Images Indistinguishable to Humans Also Indistinguishable to Classifiers?

Supplementary Material

A. Background

A.1. Diffusion model

Diffusion models [26, 52, 53] gradually inject noise into data $\boldsymbol{x}$ during the forward process:

$$\boldsymbol{z}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{1}$$

and remove noise to generate data in the reverse process. Diffusion models typically use a noise prediction network $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t)$ to predict the noise $\boldsymbol{\epsilon}$ added to $\boldsymbol{z}_t$. Noise prediction loss is defined as:

$$\mathcal{L} = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{\epsilon}}[\|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t) - \boldsymbol{\epsilon}\|_2^2]. \tag{2}$$

A.2. Frequency filter

The frequency content of an image represents the rate of pixel value changes. Low-frequency components capture overall shapes and gradual grayscale changes, while high-frequency components reflect fine details like edges and textures. Notably, convolutional neural networks can detect high-frequencies that are often imperceptible to humans [59]. In our paper, we implement a rectangular filter and discuss the choice of this filter in Appendix G.

Low-pass filters are implemented as follows: Compute the Fourier transform of the image $f(x, y)$ using $F(u, v) = \text{FFT}(f(x, y))$, and then center the spectrum with $F_c(u, v) = \text{fftshift}(F(u, v))$. Define a rectangular mask $H(u, v)$ in frequency domain, where M and N are the image dimensions:

$$H(u,v) = \begin{cases} 1 & \text{if } \left|u - \frac{M}{2}\right| \leq \text{threshold} \\ & \quad \text{and } \left|v - \frac{N}{2}\right| \leq \text{threshold}, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Apply the mask by multiplying $G(u, v) = F_c(u, v) \odot H(u, v)$, reverse the shift with $G_c(u, v) = \text{ifftshift}(G(u, v))$, and transform back to the spatial domain using $g(x, y) = \text{IFFT}(G_c(u, v))$.

High-pass filters are implemented similarly, but with the mask defined to pass high frequencies:

$$H(u,v) = \begin{cases} 0 & \text{if } \left|u - \frac{M}{2}\right| \leq \text{threshold} \\ & \quad \text{and } \left|v - \frac{N}{2}\right| \leq \text{threshold}, \\ 1 & \text{otherwise.} \end{cases} \tag{4}$$

Band-pass filters combine low-pass and high-pass filters, allowing frequencies between low and high thresholds to pass while setting others to 0.

B. Detail experiment settings

We implement our experiments upon the official code of ConvNeXt [35], U-ViT [3], DiT [42], EDM [27], EDM2 [28]. We also utilize the official models of Pixart-α [8], SDXL [43], and Playground-v2.5 [32]. The respective links and licenses are detailed in Tab. 9.

We present the main experiment settings as follows.

Dataset. For label-to-image, we consider the CIFAR [30] and ImageNet [12] datasets, which are well-established and widely recognized benchmarks in the field of image generation. For text-to-image, we utilize the COCO2014 dataset [33], known for its rich annotations and diverse image content. For CIFAR-10, we use the real CIFAR-10 training set and validation set to construct the real distribution and use the diffusion model to generate 50k training images and 10k validation images to construct the generated distribution. In the case of ImageNet, we consider 256 and 512 resolutions, which are common resolutions for ImageNet image generation tasks. We randomly sample 100k training images and 50k validation images from the training set and the validation set of ImageNet to construct the real distribution and use the diffusion models to generate 100k training images and 50k validation images to construct the generated distribution. For the real dataset, we adopt the data processing method from ADM [13] to modify the ImageNet dataset into two common resolutions: ImageNet-256 and ImageNet-512, where the numbers indicate the resolution of the data. In addition, for the COCO2014 dataset, by default, we construct the real distribution by randomly sampling 10k training images and 1k validation images from the respective training and validation sets of COCO2014. Each image in this dataset is associated with five captions. To create the generated distribution, we randomly select one of the five captions for each real image to serve as a prompt. These prompts are then used by the diffusion models to generate an equivalent number of images to construct the generated distribution.

Classifier. By default, we employ the ResNet-50 [22] as the classifier architecture. For completeness, we also consider ConvNeXt-T [35] and ViT-S [14]. Our preprocessing protocol follows the standard supervised training approach [35]. Specifically, during training, classifiers process randomly augmented crops of 224×224 images. During validation, images are resized so that their smaller dimension reaches 256 pixels while preserving the original aspect ratio. Subsequently, these images are center cropped to 224×224 pixels before being fed into the model. For the experiments on CIFAR-10, we initialize our classifier with the ResNet-

Method	Link	License
ConvNeXt	https://github.com/facebookresearch/ConvNeXt	MIT License
U-ViT	https://github.com/baofff/U-ViT	MIT License
DiT	https://github.com/facebookresearch/DiT	CC BY-NC 4.0
EDM	https://github.com/NVlabs/edm	CC BY-NC-SA 4.0
EDM2	https://github.com/NVlabs/edm2	CC BY-NC-SA 4.0
Model	**Link (add 'https://huggingface.co/')**	**License**
PixArt-α	PixArt-alpha/PixArt-XL-2-1024-MS	Open RAIL++-M
SDXL	stabilityai/stable-diffusion-xl-base-1.0	Open RAIL++-M
Playground-v2.5	playgroundai/playground-v2.5-1024px-aesthetic	Playground v2.5

Table 9. **Code links and licenses.**

50, pre-trained on ImageNet. For completeness, we also present the result trained from scratch (see experiments in Appendix. D.2). In the case of ImageNet, we opt to train the ResNet-50 model from scratch.

Diffusion model. For the generation of CIFAR-10, we consider two diffusion models: EDM [27] and U-ViT [3]. Both models have demonstrated strong generation performance on the CIFAR-10 [30]. Quantitatively, EDM achieves an FID of 1.79, and U-ViT achieves an FID of 3.11. To improve efficiency, we modified U-ViT's sampling method from Euler-Maruyama to DPM-Solver [37] and reduced the sampling steps from 1,000 to 50. These adjustments resulted in U-ViT achieving an FID of 3.65 on CIFAR-10. For the generation of ImageNet-256, we explore three diffusion models: EDM2 [28], U-ViT-H/2 [3], DiT-XL [42]. Both DiT and U-ViT are prominent diffusion transformer architectures known for their scalability and strong performance. U-ViT-H/2 achieves an FID of 2.29 on ImageNet-256, and DiT-XL/2 achieves an FID of 2.27. We consider EDM2 to incorporate a UNet-based architecture, which was traditionally used before the rise of diffusion transformers. Since EDM2 is originally designed for ImageNet-512 generation, we resize the generated images from 512 to 256 resolution to suit our ImageNet-256 task. In this way, EDM2-XXL achieves an FID of 2.14 on this task, which is similar to the FID achieved by U-ViT and DiT. For the generation of ImageNet-512, we use EDM2 [28], which achieves state-of-the-art performance on this task with an FID of 1.81. For the generation of COCO, we consider three state-of-the-art text-to-image diffusion models: Pixart-α [8], SDXL [43], and Playground-v2.5 [32].

Evaluation. We use the top-1 accuracy on the validation set to evaluate classification performance.

Training settings. The complete training settings of ResNet-50 are reported in Tab. 10 for combinations related to CIFAR-10 and Tab. 11 for combinations related to ImageNet.

C. Computational cost

Our experiments were conducted on RTX 3090 and V100 GPUs. The detailed computational costs are presented in Tab. 12. Training epochs were set to 50 for CIFAR-10 and 200 for ImageNet. The number of epochs trained on CIFAR-10 is relatively low because we use a pre-trained model to initialize our classifier, enabling faster convergence.

D. Additional results

D.1. Additional results from other generative models

As shown in Tab. 13, we conducted experiments using StyleGAN-XL [49], a state-of-the-art GAN model trained on ImageNet-256, as well as SiT [39], a flow matching model that extends the applicability of the method. Our results demonstrate that, despite the use of a discriminator during GAN training, the classifier can still easily distinguish between real and generated images. We argue that this is because the discriminator is trained jointly with the generator. During training, the generated data seen by the discriminator comes from a continuously evolving distribution, as the generator improves with each iteration. However, when using a classifier to distinguish between real and generated distributions, the generated distribution remains fixed.

Our experiments confirm that images generated by GANs can also be readily distinguished from real images using a classifier, despite GANs' adversarial training approach. We have not yet conducted experiments to explore this phenomenon in other generative frameworks such as Masked Image Generation models [7, 66] or Autoregressive models [54, 55], which remains an interesting direction for future work.

D.2. Results of CIFAR-10

In order to ensure the completeness of the experiment, we are here to present the result of CIFAR-10 trained from scratch. We present the results in Tab. 14. If there is no prior

Config	Value
Optimizer	AdamW
Learning rate	4e-4
Weight decay	0.05
Optimizer momentum	β_1, β_2=0.9, 0.999
Batch size	256
Learning rate schedule	Cosine decay
Warmup epochs	0
Training epochs	50
Augmentation	RandAug (9, 0.5)
Label smoothing	0.1
Mixup	0
Cutmix	0

Table 10. **Training settings for CIFAR-10.**

Config	Value
Optimizer	AdamW
Learning rate	1e-3
Weight decay	0.3
Optimizer momentum	β_1, β_2=0.9, 0.95
Batch size	4096
Learning rate schedule	Cosine decay
Warmup epochs	20
Training epochs	200
Augmentation	RandAug (9, 0.5)
Label smoothing	0.1
Mixup	0.8
Cutmix	1.0

Table 11. **Training settings for ImageNet.**

Model	Combinations	Epochs	GPU-type	GPU-nums	Hours
ResNet-50	*C*, *U*	50	3090	4	2
ViT-S	*C*, *U*	50	3090	4	2
ConvNeXt-T	*C*, *U*	50	3090	4	2
ResNet-50	*I*-256, *U*-H/2	200	V100	8	6
ViT-S	*I*-256, *U*-H/2	200	V100	8	9
ConvNeXt-T	*I*-256, *U*-H/2	200	V100	8	8
ResNet-50	*I*-512, *E*2-XXL	200	V100	8	9
ViT-S	*I*-512, *E*2-XXL	200	V100	8	11
ConvNeXt-T	*I*-512, *E*2-XXL	200	V100	8	10

Table 12. **Training time of classifiers**.

knowledge, classifiers struggle to distinguish between real and generated data in datasets with low resolution, such as CIFAR-10 (i.e., 32x32). However, the use of a pre-trained model allows the features learned from the ImageNet dataset to aid in differentiating between real and generated images in CIFAR-10.

D.3. Self-supervised classifiers for Text-to-Image distribution classification

As shown in Tab. 15, we report the distribution classification accuracy of self-supervised classifiers in text-to-image scenarios. Notably, these classifiers achieve high accuracy in distinguishing between different text-to-image models. This suggests that there are significant differences between text-to-image generative models, allowing even self-supervised classifiers to easily distinguish them.

D.4. Visualization of crops

As shown in Fig. 7, we present the visualization of crops mentioned in Sec. 6.1.

D.5. Frequency analysis on the combination of EDM2-XS and EDM2-XXL

We preprocess the generated images using band-pass filters as defined in Sec. A.2, with four threshold intervals: 0-10, 10-30, 30-50, and 50-100. An example of original and processed EDM2-XXL generated images is shown in Fig. 8. As shown in Tab. 16 and Fig. 9, for EDM2-XS and EDM2-XXL, the smallest and largest models in the EDM2 family, classifiers' accuracy approaches random guessing (around 50%) across different threshold intervals. This indicates that for models within the same diffusion model family, which share inductive biases but differ in visual quality, classifiers unable to distinguish between them based on any specific frequency band.

D.6. User study

Fig. 10 shows a screenshot of the interaction interface used in our user study. The study involved nineteen participants, and we designed three groups of experiments, each requiring participants to classify 32 pairs of images. In the first set, participants distinguished between generated and real images, with real images sourced from ImageNet-256 and

Real dataset	Generative model	FID	Classifier	Accuracy (%)
I-256	StyleGAN-XL [49]	2.30	ResNet-50	99.69
			ViT-S	99.95
			ConvNeXt-T	96.85
I-256	SiT-XL [39]	2.06	ResNet-50	99.87

Table 13. **Binary distribution classification on label-to-image**. All classifiers yield high accuracy on various datasets against strong generative models. FIDs are taken from the corresponding references.

Real dataset	Generative model	Model	Pretrained	Scratch
C	*E* [27]	ResNet-50	96.25	56.13
		ViT-S	89.38	55.32
		ConvNeXt-T	98.43	53.27
C	*U* [3]	ResNet-50	99.92	56.70
		ViT-S	98.04	52.66
		ConvNeXt-T	99.96	56.37

Table 14. **Distribution classification accuracy on CIFAR-10.** "Pretrained" indicates that the classifier was initialized with a model pretrained on ImageNet, while "Scratch" indicates that the classifier was trained from scratch.

generated images from U-ViT-H/2. In the second set, they classified images between two diffusion models with similar performance: DiT-XL/2 and U-ViT-H/2. For the final set, participants evaluated images from EDM2-XS and EDM2-XXL, which share the same training methodology but differ in parameter count, resulting in different FID scores and visual quality. In the first set, participants were asked to identify the real images. In the second, they were tasked with identifying DiT images, and reference images from DiT and U-ViT were provided during the test. In the third experiment, participants judged which images were of higher quality. All nineteen participants were graduate students with substantial experience in machine learning. They were allowed to zoom in on the images during the experiment, which was conducted on 27-inch 4K displays. All participants had corrected vision of 1.0 (standard normal vision), and their ages ranged from 22 to 26. Each participant completed the experiment within an hour and was compensated $10.

E. Binary classification as a measure of distribution distance

We employ a classifier $C(x)$ to distinguish between the real data distribution $p_{\text{data}}(x)$ and the generated data distribution $p_g(x)$. By training $C(x)$ using the binary cross-entropy loss:

$$L(C) = -E_{x\sim p_{\text{data}}(x)}[\log(C(x))] - E_{x\sim p_g(x)}[\log(1-C(x))], \tag{5}$$

The optimal classifier that minimizes this loss is:

$$C^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_g(x)}. \tag{6}$$

Substituting $C^*(x)$ back into the loss function yields:

$$L(C^*) = -\log(4) + 2\,\text{JSD}(p_{\text{data}}(x)\|p_g(x)), \tag{7}$$

where JSD denotes the Jensen-Shannon Divergence—a direct measure of the distance between the two distributions.

In our paper, we use classification accuracy to evaluate how well the classifier distinguishes between real and generated data because it provides an intuitive and interpretable metric. Although accuracy is non-differentiable and unsuitable for direct optimization, training the classifier with the binary cross-entropy loss—a convex surrogate—often leads to improved accuracy. This correlation suggests that accuracy can serve as a proxy for changes in the loss function, reflecting the distance between the generated and real data distributions.

F. Relationship between distribution classification and FID

The Fréchet Inception Distance (FID) is a widely used metric for evaluating the quality of generative models. It relies on feature extraction networks trained on datasets such as ImageNet and assumes that the extracted feature vectors follow a multivariate Gaussian distribution. FID calculates the Fréchet distance between these Gaussian distributions to measure discrepancies between the real and generated data.

Training samples	Self-supervised method	Accuracy (%)
5k	MAE	87.77
5k	MoCo v3	90.48
10k	MAE	91.67
10k	MoCo v3	90.90

Table 15. **Four-way distribution classification on text-to-image**. Linear classifier on self-supervised features also achieves high accuracy in distinguishing four distributions: COCO [33], Pixart-α [8], SDXL [43], and Playground-v2.5 [32], using only 5k or 10k training samples per dataset. We use ViT-B as the backbone.

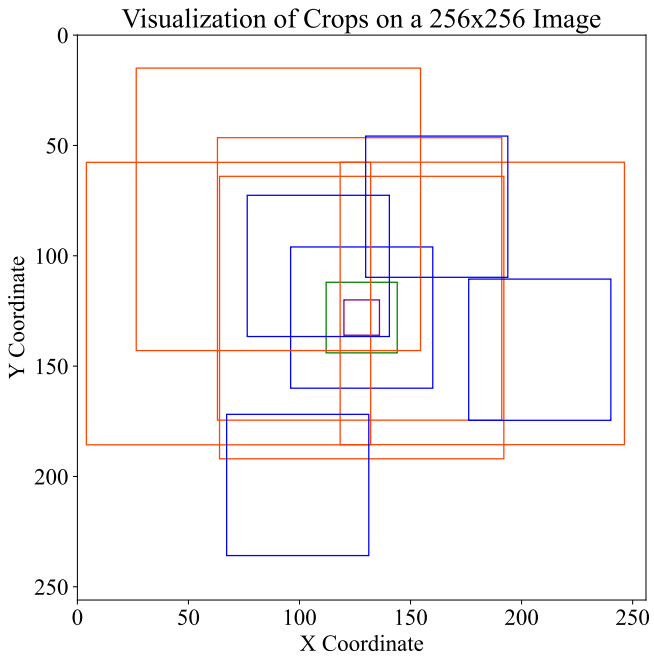


Figure 7. **Visualization of crops.** With each resolution represented by a different color.

Combinations	Classifier	Intervals	Accuracy
*E*2-XS, *E*2-XXL	ResNet-50	0-10	57.94
		10-30	58.96
		30-50	58.12
		50-100	50.48

Table 16. **Classification accuracy** of ResNet-50 on the combination of EDM2-XS and EDM2-XXL after applying band-pass filters.

Meanwhile, as noted by Kynkäänniemi et al. [31], FID can decrease simply by aligning the histograms of top-N classifications, without necessarily improving the perceptual quality of the generated images. Additionally, recent works like Karras et al. [28] and Tian et al. [55] report FID scores close to those of the ImageNet validation set, suggesting limitations in FID's sensitivity to certain distribution differences.

In contrast, our classifier-based approach offers a more direct measure of the distance between distributions without relying on the Gaussian assumptions inherent in FID. By training a classifier to distinguish between real and generated data, we obtain an intuitive and interpretable metric that reflects the actual distribution differences. This method complements commonly used metrics such as FID and Inception Score (IS), providing an alternative perspective on evaluating generative models.

G. Comparison of rectangular and circular filters

Rectangular and circular filters are common techniques in image filtering. In this paper, we chose to implement a rectangular filter following the official FreeU implementation [51], due to its simplicity and computational efficiency. For comparison, Fig. 11 presents initial results using a circular mask, denoted as U-H/2 and I-256 (Ideal). The results from both implementations are similar, and we chose to use

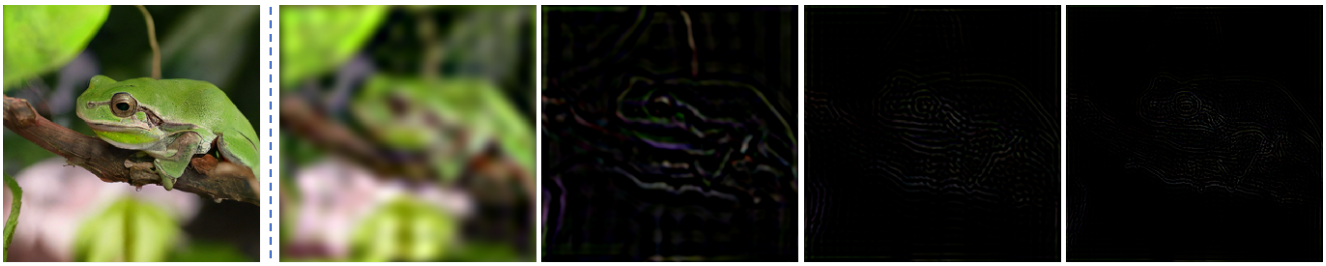


Figure 8. **Visualization of frequency domain processing of EDM2-XXL.** The image shows the results after applying a band-pass filter to EDM2-XXL with band thresholds of 0-10, 10-30, 30-50, and 50-100, from left to right.

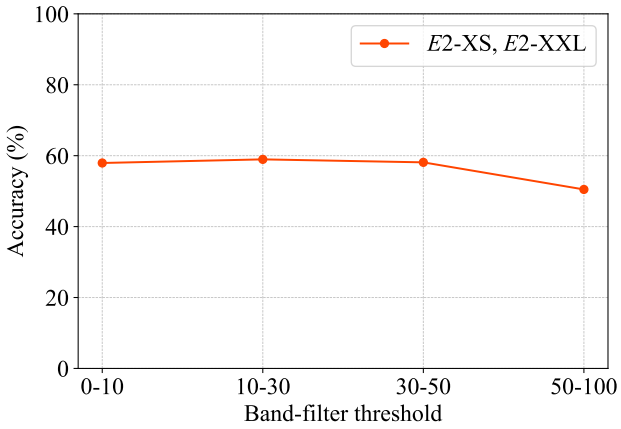
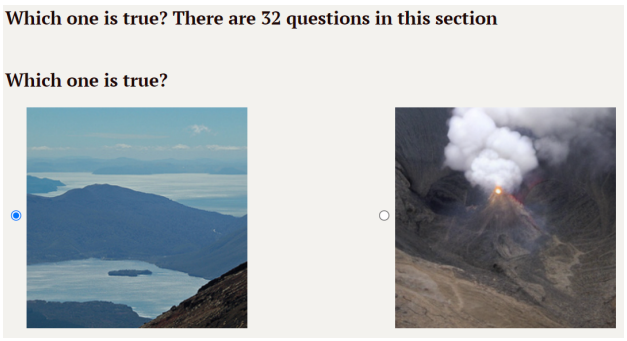
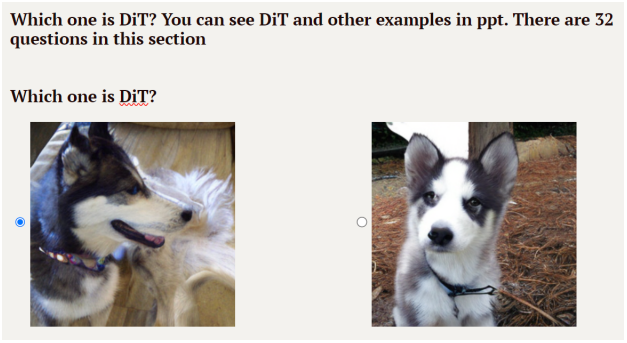


Figure 9. **Accuracy vs. band-frequency filter threshold**

the rectangular filter for its computational efficiency.



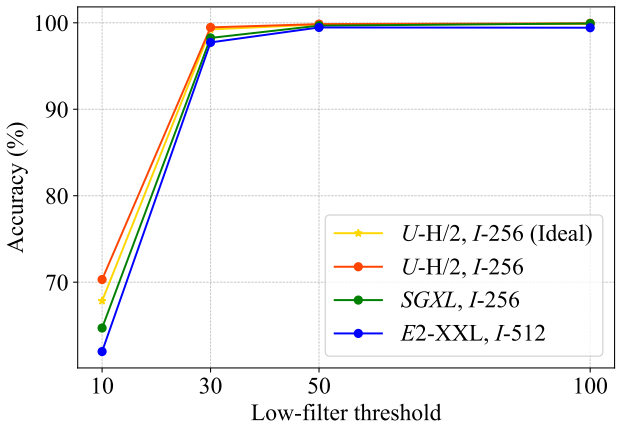
(a) *I*-256 vs. *U*-ViT-H/2



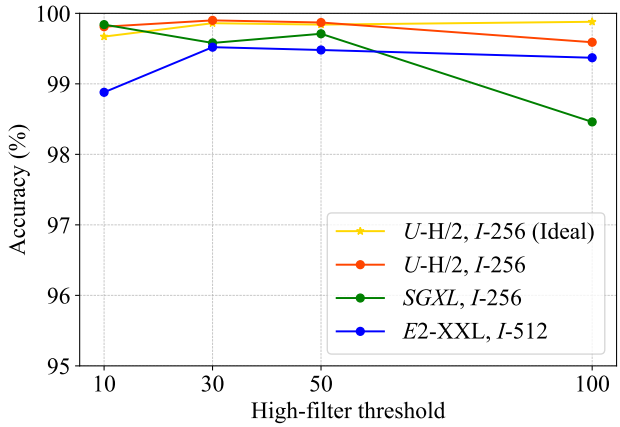
(b) *U*-ViT-H/2 vs. *D*

(c) *E*2-XS vs. *E*2-XXL

Figure 10. **Screenshot of user study.** Participants are asked to distinguish generated distributions from real ones and to classify which diffusion model generated a given image. Each group of experiments is illustrated with a separate example here. In each set of experiments, we also randomized the order to prevent examples from the same set from influencing each other.



(a) Accuracy vs. low-frequency filter threshold



(b) Accuracy vs. high-frequency filter threshold

Figure 11. Comparison of model accuracy across different filter thresholds using rectangular and circular filters.