

Training Data Provenance Verification: Did Your Model Use Synthetic Data from My Generative Model for Training?

Supplementary Material

A. Proofs

Theorem 1. Assume M is trained on synthetic data generated by a text-to-image model G , and a set of text prompts \mathcal{T}_1 , i.e., $P_1(\mathbf{x}) = P(\mathbf{x}|G, \mathcal{T}_1)$. \hat{M} can be trained on either real data or synthetic data generated by any text-to-image model. Based on the generalization errors of M and \hat{M} on the target domain $T = \{\mathcal{X}, P(\mathbf{x}|G, \mathcal{T}_t)\}$, we have:

$$\sup_{P_2(\mathbf{x})=P(\mathbf{x}|G, \mathcal{T}_2)} |\Delta\epsilon_T| \leq \sup_{P_2(\mathbf{x}) \perp G} |\Delta\epsilon_T|, \quad (3)$$

where $\Delta\epsilon_T$ represents the difference in generalization error between M and \hat{M} on the target domain T , expressed as $\Delta\epsilon_T = \epsilon_T(M) - \epsilon_T(\hat{M})$. $P_2(\mathbf{x}) \perp G$ denotes that $P_2(\mathbf{x})$ is independent of G , meaning $P_2(\mathbf{x}) = P(\mathbf{x}|G', \mathcal{T}_2)$ or $P_2(\mathbf{x}) = P_R(\mathbf{x})$, where G' is a text-to-image model different from G and $P_R(\mathbf{x})$ represents the distribution of real data. \mathcal{T}_1 , \mathcal{T}_2 and \mathcal{T}_t are distinct sets of text prompts.

Proof. Theorem 1 in [2] states the upper bound of the generalization error of model M' on the target domain T . Let \mathcal{H} be a hypothesis space with VC dimension d' , and let m' be the sample size of the dataset in the source domain S . Then, with probability at least $1 - \delta$, for every $M' \in \mathcal{H}$:

$$\epsilon_T(M') \leq \hat{\epsilon}_S(M') + \sqrt{\frac{4}{m'} \left(d' \log \frac{2em'}{d'} \right)} + d_{\mathcal{H}}(P_S, P_T) + \lambda, \quad (4)$$

where $\epsilon_T(M')$ and $\hat{\epsilon}_S(M')$ represent the generalization error of M' on the target domain and the empirical error of M' on the source domain, respectively. P_S and P_T denote the marginal probability distributions of the source and target domains, respectively. $d_{\mathcal{H}}(P_S, P_T)$ is the \mathcal{H} -divergence between the source and target domains, which measures the similarity between the two distributions (source and target) within the hypothesis space \mathcal{H} . A smaller $d_{\mathcal{H}}(P_S(\mathbf{x}), P_T(\mathbf{x}))$ indicates that the distributions of the source and target domains are closer. λ is a constant and e is the base of the natural logarithm.

Substitute $P_T = P(\mathbf{x}|G, \mathcal{T}_t)$ into Eq. (4), for M with the source domain distribution $P(\mathbf{x}|G, \mathcal{T}_1)$:

$$\epsilon_T(M) \leq \hat{\epsilon}_S(M) + \sqrt{\frac{4}{m} \left(d \log \frac{2em}{d} \right)} + d_{\mathcal{H}}(P(\mathbf{x}|G, \mathcal{T}_1), P(\mathbf{x}|G, \mathcal{T}_t)) + \lambda \quad (5)$$

For \hat{M} with the source domain distribution $P(\mathbf{x}|G, \mathcal{T}_2)$,

similarly:

$$\epsilon_T(\hat{M}_1) \leq \hat{\epsilon}_S(\hat{M}_1) + \sqrt{\frac{4}{\hat{m}_1} \left(\hat{d} \log \frac{2e\hat{m}_1}{\hat{d}} \right)} + d_{\mathcal{H}}(P(\mathbf{x}|G, \mathcal{T}_2), P(\mathbf{x}|G, \mathcal{T}_t)) + \lambda \quad (6)$$

To distinguish from later cases, we denote \hat{M} trained in this situation as \hat{M}_1 and m as \hat{m}_1 . For \hat{M} with the source domain distribution $P(\mathbf{x}|G', \mathcal{T}_2)$, similarly:

$$\epsilon_T(\hat{M}_2) \leq \hat{\epsilon}_S(\hat{M}_2) + \sqrt{\frac{4}{\hat{m}_2} \left(\hat{d} \log \frac{2e\hat{m}_2}{\hat{d}} \right)} + d_{\mathcal{H}}(P(\mathbf{x}|G', \mathcal{T}_2), P(\mathbf{x}|G, \mathcal{T}_t)) + \lambda \quad (7)$$

We denote \hat{M} trained in this situation as \hat{M}_2 and m as \hat{m}_2 .

We denote the upper bounds in Eq. (5), Eq. (6), and Eq. (7) as ξ , $\hat{\xi}_1$, and $\hat{\xi}_2$, respectively. Then we calculate the upper bound of $|\epsilon_T(M) - \epsilon_T(\hat{M}_1)|$:

$$\sup |\epsilon_T(M) - \epsilon_T(\hat{M}_1)| = \max\{\xi, \hat{\xi}_1\} \quad (8)$$

For the upper bound of $|\epsilon_T(M) - \epsilon_T(\hat{M}_2)|$:

$$\sup |\epsilon_T(M) - \epsilon_T(\hat{M}_2)| = \max\{\xi, \hat{\xi}_2\} \quad (9)$$

Assuming that \hat{M} is trained on the same amount of source domain data in both cases, i.e., $\hat{m}_1 = \hat{m}_2$. Additionally, assuming the empirical errors on the training set are also the same, i.e., $\hat{\epsilon}_S(\hat{M}_1) = \hat{\epsilon}_S(\hat{M}_2)$. Therefore, the only difference between $\hat{\xi}_1$ and $\hat{\xi}_2$ lies in the $d_{\mathcal{H}}(\cdot, \cdot)$ term. Since the generative model in the conditions for $P(\mathbf{x}|G, \mathcal{T}_2)$ and $P(\mathbf{x}|G, \mathcal{T}_t)$ is the same while the text conditions differ, whereas both differ in the conditions for $P(\mathbf{x}|G', \mathcal{T}_2)$ and $P(\mathbf{x}|G, \mathcal{T}_t)$, it follows that:

$$d_{\mathcal{H}}(P(\mathbf{x}|G, \mathcal{T}_2), P(\mathbf{x}|G, \mathcal{T}_t)) \leq d_{\mathcal{H}}(P(\mathbf{x}|G', \mathcal{T}_2), P(\mathbf{x}|G, \mathcal{T}_t)) \quad (10)$$

Therefore, $\hat{\xi}_1 \leq \hat{\xi}_2$, i.e.,

$$\sup |\epsilon_T(M) - \epsilon_T(\hat{M}_1)| \leq \sup |\epsilon_T(M) - \epsilon_T(\hat{M}_2)| \quad (11)$$

Alternatively, it can be expressed as:

$$\sup_{P_2(\mathbf{x})=P(\mathbf{x}|G, \mathcal{T}_2)} |\Delta\epsilon_T| \leq \sup_{P_2(\mathbf{x})=P(\mathbf{x}|G', \mathcal{T}_2)} |\Delta\epsilon_T|, \quad (12)$$

where $\Delta\epsilon_T = \epsilon_T(M) - \epsilon_T(\hat{M})$.

Similarly, when \hat{M} is trained on real data $\mathbf{x} \sim P_R(\mathbf{x})$, we have:

$$\sup_{P_2(\mathbf{x})=P(\mathbf{x}|G, T_2)} |\Delta\epsilon_T| \leq \sup_{P_2(\mathbf{x})=P_R(\mathbf{x})} |\Delta\epsilon_T| \quad (13)$$

By combining Eq. (12) and Eq. (13), we obtain:

$$\sup_{P_2(\mathbf{x})=P(\mathbf{x}|G, T_2)} |\Delta\epsilon_T| \leq \sup_{P_2(\mathbf{x}) \perp G} |\Delta\epsilon_T|, \quad (14)$$

□

B. The Details of Experiments

B.1. Datasets Used

CIFAR10 [20]: The CIFAR10 dataset consists of 32x32 colored images with 10 classes. There are 50000 training images and 10000 test images.

CIFAR100 [20]: The CIFAR100 dataset consists of 32x32 coloured images with 100 classes. There are 50000 training images and 10000 test images.

ImageNet-100 [40]: A randomly chosen subset of ImageNet-1K [5], which has larger sized coloured images with 100 classes. There are approximately 126,689 training images and 5000 test images. As is commonly done, we resize all images to be of size 224x224. The specific categories we use are listed in ImageNet-100.txt in the supplementary material.

B.2. Pre-trained Text-to-image Models Used

We use four pre-trained text-to-image models, as follows:

Stable Diffusion v1.4 [38]: Stable Diffusion v1.4 is trained on 512x512 images from a subset of the LAION-5B [41] dataset. This model uses a frozen CLIP ViT-L/14 text encoder to condition the model on text prompts. Its pre-trained weights can be obtained from <https://huggingface.co/CompVis/stable-diffusion-v1-4>.

Latent Consistency Model [28]: Latent consistency model is trained on 768x768 images from a subset of the LAION-5B [41] dataset, named LAION-Aesthetics. Its pre-trained weights can be obtained from https://huggingface.co/SimianLuo/LCM_Dreamshaper_v7.

Pixart- α [4]: Pixart- α is trained on images from SAM [19] dataset, whose text prompts are generated by LLaVA [23]. Its pre-trained weights can be obtained from <https://huggingface.co/PixArt-alpha/PixArt-XL-2-512x512>.

Stable Cascade [34]: Stable Cascade is built upon the Würstchen architecture [34], and trained on images from a subset of the LAION-5B [41] dataset.

Its pre-trained weights can be obtained from <https://huggingface.co/stabilityai/stable-cascade>.

B.3. Parameter Setting Details

- **Experiments conducted on CIFAR10/CIFAR100:** We use the methods in [42] to generate 60,000 images as CIFAR10-Syn and CIFAR100-Syn for training the suspicious models. Specifically, we generate 20,000 images each using the techniques of “Class Prompt”, “Multi-Domain”, and “Random Unconditional Guidance”. Additionally, the shadow dataset and validation dataset retain default parameters except for the text prompts and resolution. Focal loss with $\gamma = 2$ and $\alpha = 0.25$. All models are trained using the AdamW optimizer.
- **Experiments conducted on ImageNet-100:** We use the methods in [40] to generate 100,000 images as ImageNet-100-Syn for training the suspicious models. Specifically, we use “ c, d_c ” to generate these images, where c is the class name, and d_c refers to the definition of class c provided by WordNet [30]. Guidance scale is 2. Additionally, the shadow dataset and validation dataset retain default parameters except for the text prompts and resolution. Focal loss with $\gamma = 2$ and $\alpha = 0.25$. All models are trained using the AdamW optimizer.

B.4. The Details of Han *et al.*’s Work

Unlike our setup, the work by Han *et al.* assumes that the defender has access to multiple different generative models to train an attributor. To enable comparison with this method, we followed this setup when using it. Specifically, we assume that the defender, in addition to having G_d , also possesses the other three text-to-image models mentioned in the paper (whereas in our setup, the defender has only one text-to-image model, G_d). Using these generative models, we create four shadow datasets following the same approach, as well as one validation dataset generated using G_d in the same manner.

Based on each shadow dataset, we train 16 shadow models, all of which are ResNet18 with varied training hyperparameters. In total, we obtain 64 shadow models. Using these shadow models, we infer logits on the validation dataset to train an attributor. The attributor is a two-layer fully connected network with an input dimension matching the dimension of the logits and an output dimension of 2. It classifies logits of shadow models trained on synthetic data generated by G_d as 1 and others as 0. The attributor is trained for 10 epochs.

B.5. The Details of Interference Resistance Experiments

Multiple training data sources for M_{sus} . In this experiment, the methods for generating the synthetic dataset and

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
PixArt- α [4]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Stable Cascade [34]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	PixArt- α			32	32			
	Real Data			32	32			
Average Value						0.500	0.286	0.500

Table A1. The results of random classification on CIFAR10.

training the network follow the settings in Sec. 4.1, with the only difference being that the training dataset for the suspicious model consists of both the synthetic and real datasets.

Fine-tuning G_d . In this experiment, we fine-tune G_d using LoRA with a batch size of 16, a learning rate of 1e-4 and a resolution of 512. The methods for generating dataset and training network follow the settings in Sec. 4.1.

Fine-tuning M_{sus} . In this experiment, we fine-tune the suspicious model with a learning rate of 1e-5, a weight decay of 1e-4, a batch size of 64, and the cross-entropy loss function. The methods for generating the synthetic dataset and training the network follow the settings in Sec. 4.1.

B.6. Hypothesis Testing

We represent the generalization errors of M_{sdw} and M_{sus} by the sets of accuracies \mathcal{A}_{sdw} and \mathcal{A}_{sus} for each batch on the validation dataset. Then we use the Grubbs test to determine whether the mean of \mathcal{A}_{sus} is a low-value outlier of \mathcal{A}_{sdw} . The pseudocode is shown in Algorithm 1, where $t_{1-\frac{\alpha}{n}, n-2}$ is the critical value of the t-distribution with degrees of freedom $n - 2$ and significance level α . When $G > G_0$, we consider $mean(\mathcal{A}_{sus})$ to be a low-value outlier of \mathcal{A}_{sdw} , meaning the generalization error gap between two models is significant, and M_{sus} is deemed legitimate. When $G \leq G_0$, $mean(\mathcal{A}_{sus})$ falls within the distribution of \mathcal{A}_{sdw} , indicating that the generalization errors of two models are close, and M_{sus} is deemed illegal.

C. Results on CLIP

We fine-tuned CLIP [35] on ImageNet-100 using hyperparameters in Sec. 4.1, with ResNet50 and ViT-B. Then We calculate accuracies of TrainProVe and Han *et al.*'s work on

Algorithm 1 Grubbs' Hypothesis Test

- 1: **Input:** $\mathcal{A}_{sdw}, \mathcal{A}_{sus}$, significance level $\alpha = 0.05$
- 2: $m_{sdw} = mean(\mathcal{A}_{sdw}), s_{sdw} = std(\mathcal{A}_{sdw})$
- 3: $n = len(\mathcal{A}_{sdw}), m_{sus} = mean(\mathcal{A}_{sus})$
- 4: $G = \frac{m_{sdw} - m_{sus}}{s_{sdw}}, G_0 = \frac{(n-1)}{\sqrt{n}} \sqrt{\frac{(t_{1-\frac{\alpha}{n}, n-2})^2}{n-2 + (t_{1-\frac{\alpha}{n}, n-2})^2}}$
- 5: **Output:** $G > G_0$

Stable Diffusion v1.4. Finally, the accuracy of TrainProVe reached 0.75, while Han *et al.*'s work only achieved 0.56, meaning that compared to the baseline, TrainProVe can still be applied to more complex scenarios.

D. More Experimental Results

Here, we present the specific results of different methods on various datasets (as shown in Tab. 1 of the paper), as shown in Tab. A1 - Tab. A14.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			0	64			
	PixArt- α	60	4	0	64	0.756	0.606	0.824
	Stable Cascade			14	50			
	Real Data			60	4			
Latent Consistency Model [28]	Stable Diffusion v1.4			4	60			
	PixArt- α	64	0	0	64	0.969	0.928	0.980
	Stable Cascade			2	62			
	Real Data			4	60			
PixArt- α [4]	Stable Diffusion v1.4			0	64			
	Latent Consistency Model	47	17	0	64	0.919	0.783	0.850
	Stable Cascade			9	55			
	Real Data			0	64			
Stable Cascade [34]	Stable Diffusion v1.4			60	4			
	Latent Consistency Model	0	64	53	11	0.428	0.000	0.268
	PixArt- α			0	64			
	Real Data			6	58			
Average Value						0.768	0.579	0.731

Table A2. The results of Han *et al.*'s method on CIFAR10.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			4	60			
	PixArt- α	64	0	4	60	0.941	0.871	0.963
	Stable Cascade			7	57			
	Real Data			4	60			
Latent Consistency Model [28]	Stable Diffusion v1.4			0	64			
	PixArt- α	22	42	0	64	0.869	0.512	0.672
	Stable Cascade			0	64			
	Real Data			0	64			
PixArt- α [4]	Stable Diffusion v1.4			25	39			
	Latent Consistency Model	64	0	32	32	0.609	0.506	0.756
	Stable Cascade			64	0			
	Real Data			4	60			
Stable Cascade [34]	Stable Diffusion v1.4			17	47			
	Latent Consistency Model	64	0	5	59	0.828	0.699	0.893
	PixArt- α			19	45			
	Real Data			14	50			
Average Value						0.812	0.647	0.821

Table A3. The results of TrainProVe-Sim on CIFAR10.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			1	63			
	PixArt- α	57	7	0	64	0.928	0.832	0.914
	Stable Cascade			11	53			
	Real Data			4	60			
Latent Consistency Model [28]	Stable Diffusion v1.4			0	64			
	PixArt- α	60	4	0	64	0.988	0.968	0.969
	Stable Cascade			0	64			
	Real Data			0	64			
PixArt- α [4]	Stable Diffusion v1.4			9	55			
	Latent Consistency Model	64	0	22	42	0.725	0.593	0.828
	Stable Cascade			54	10			
	Real Data			3	61			
Stable Cascade [34]	Stable Diffusion v1.4			9	55			
	Latent Consistency Model	64	0	16	48	0.831	0.703	0.895
	PixArt- α			23	41			
	Real Data			6	58			
Average Value						0.868	0.774	0.902

Table A4. The results of TrainProVe-Ent on CIFAR10.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
PixArt- α [4]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Stable Cascade [34]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	PixArt- α			32	32			
	Real Data			32	32			
Average Value						0.500	0.286	0.500

Table A5. The results of random classification on CIFAR100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			0	64			
	PixArt- α	64	0	9	55	0.772	0.637	0.861
	Stable Cascade			32	32			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			0	64			
	PixArt- α	64	0	0	64	1.000	1.000	1.000
	Stable Cascade			0	64			
	Real Data			0	64			
PixArt- α [4]	Stable Diffusion v1.4			0	64			
	Latent Consistency Model	36	28	0	64	0.791	0.518	0.705
	Stable Cascade			32	32			
	Real Data			7	57			
Stable Cascade [34]	Stable Diffusion v1.4			28	36			
	Latent Consistency Model	4	60	64	0	0.356	0.037	0.246
	PixArt- α			47	17			
	Real Data			7	57			
Average Value						0.730	0.548	0.703

Table A6. The results of Han *et al.*'s Method on CIFAR100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			32	32			
	PixArt- α	28	36	29	35	0.509	0.263	0.498
	Stable Cascade			32	32			
	Real Data			28	36			
Latent Consistency Model [28]	Stable Diffusion v1.4			12	52			
	PixArt- α	32	32	12	52	0.672	0.379	0.607
	Stable Cascade			22	42			
	Real Data			10	54			
PixArt- α [4]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Stable Cascade [34]	Stable Diffusion v1.4			16	48			
	Latent Consistency Model	32	32	32	32	0.619	0.344	0.574
	PixArt- α			26	38			
	Real Data			16	48			
Average Value						0.575	0.318	0.545

Table A7. The results of TrainProVe-Sim on CIFAR100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			64	0			
	PixArt- α	64	0	55	9	0.347	0.380	0.592
	Stable Cascade			58	6			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			25	39			
	PixArt- α	64	0	32	32	0.628	0.518	0.768
	Stable Cascade			48	16			
	Real Data			14	50			
PixArt- α [4]	Stable Diffusion v1.4			60	4			
	Latent Consistency Model	64	0	64	0	0.253	0.349	0.533
	Stable Cascade			64	0			
	Real Data			51	13			
Stable Cascade [34]	Stable Diffusion v1.4			41	23			
	Latent Consistency Model	64	0	62	2	0.363	0.386	0.602
	PixArt- α			64	0			
	Real Data			37	27			
Average Value						0.398	0.408	0.624

Table A8. The results of TrainProVe-Ent on CIFAR100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			0	64			
	PixArt- α	64	0	0	64	0.988	0.970	1.000
	Stable Cascade			0	64			
	Real Data			0	64			
Latent Consistency Model [28]	Stable Diffusion v1.4			0	64			
	PixArt- α	64	0	0	64	1.000	1.000	1.000
	Stable Cascade			0	64			
	Real Data			0	64			
PixArt- α [4]	Stable Diffusion v1.4			0	64			
	Latent Consistency Model	55	9	0	64	0.972	0.924	0.930
	Stable Cascade			0	64			
	Real Data			0	64			
Stable Cascade [34]	Stable Diffusion v1.4			0	64			
	Latent Consistency Model	63	1	0	64	0.997	0.992	0.992
	PixArt- α			0	64			
	Real Data			0	64			
Average Value						0.992	0.979	0.981

Table A9. The results of TrainProVe on CIFAR100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			32	32			
	PixArt- α	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
PixArt- α [4]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	Stable Cascade			32	32			
	Real Data			32	32			
Stable Cascade [34]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	32	32	0.500	0.286	0.500
	PixArt- α			32	32			
	Real Data			32	32			
Average Value						0.500	0.286	0.500

Table A10. The results of random classification on ImageNet-100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			31	33			
	PixArt- α	32	32	0	64	0.684	0.388	0.615
	Stable Cascade			6	58			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			15	49			
	PixArt- α	26	38	6	58	0.772	0.416	0.635
	Stable Cascade			1	63			
	Real Data			13	51			
PixArt- α [4]	Stable Diffusion v1.4			24	40			
	Latent Consistency Model	64	0	3	61	0.753	0.618	0.846
	Stable Cascade			32	32			
	Real Data			20	44			
Stable Cascade [34]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	32	32	40	24	0.475	0.276	0.484
	PixArt- α			32	32			
	Real Data			32	32			
Average Value						0.671	0.425	0.645

Table A11. The results of Han *et al.*'s Method on ImageNet-100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			32	32			
	PixArt- α	32	32	30	34	0.506	0.288	0.504
	Stable Cascade			32	32			
	Real Data			32	32			
Latent Consistency Model [28]	Stable Diffusion v1.4			16	48			
	PixArt- α	28	36	16	48	0.641	0.327	0.564
	Stable Cascade			26	38			
	Real Data			21	43			
PixArt- α [4]	Stable Diffusion v1.4			28	36			
	Latent Consistency Model	32	32	32	32	0.519	0.294	0.512
	Stable Cascade			32	32			
	Real Data			30	34			
Stable Cascade [34]	Stable Diffusion v1.4			26	38			
	Latent Consistency Model	32	32	31	33	0.553	0.309	0.533
	PixArt- α			28	36			
	Real Data			26	38			
Average Value						0.555	0.305	0.528

Table A12. The results of TrainProVe-Sim on ImageNet-100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			53	11			
	PixArt- α	64	0	60	4	0.266	0.353	0.541
	Stable Cascade			62	2			
	Real Data			60	4			
Latent Consistency Model [28]	Stable Diffusion v1.4			41	23			
	PixArt- α	51	13	48	16	0.441	0.363	0.574
	Stable Cascade			37	27			
	Real Data			40	24			
PixArt- α [4]	Stable Diffusion v1.4			56	8			
	Latent Consistency Model	60	4	41	23	0.338	0.361	0.563
	Stable Cascade			59	5			
	Real Data			52	12			
Stable Cascade [34]	Stable Diffusion v1.4			53	11			
	Latent Consistency Model	60	4	36	28	0.381	0.377	0.590
	PixArt- α			53	11			
	Real Data			52	12			
Average Value						0.356	0.364	0.567

Table A13. The results of TrainProVe-Ent on ImageNet-100.

The Suspect's Data Sources		TP	FP	FN	TN	Accuracy	F1 Score	AUROC
The Defender's Generative Model G_d	Data Sources Unrelated to G_d							
Stable Diffusion [38] v1.4	Latent Consistency Model			0	64			
	PixArt- α	64	0	0	64	0.819	0.688	0.887
	Stable Cascade			11	53			
	Real Data			47	17			
Latent Consistency Model [28]	Stable Diffusion v1.4			0	64			
	PixArt- α	2	62	0	64	0.806	0.061	0.517
	Stable Cascade			0	64			
	Real Data			0	64			
PixArt- α [4]	Stable Diffusion v1.4			29	35			
	Latent Consistency Model	64	0	0	64	0.734	0.601	0.834
	Stable Cascade			24	40			
	Real Data			32	32			
Stable Cascade [34]	Stable Diffusion v1.4			32	32			
	Latent Consistency Model	59	5	0	64	0.784	0.631	0.836
	PixArt- α			0	64			
	Real Data			32	32			
Average Value						0.786	0.495	0.769

Table A14. The results of TrainProVe on ImageNet-100.