ArtistAuditor: **Auditing Artist Style Pirate in Text-to-image Generation Models**

Anonymous Author(s)

Abstract

The text-to-image models based on diffusion processes, such as DALL-E, Stable Diffusion, and Midjourney, are capable of transforming texts into detailed images and have widespread applications in art and design. As such, amateur users can easily imitate professional-level paintings by collecting an artist's work and fine-tuning the model, leading to concerns about artworks' copyright infringement. To tackle these issues, previous studies either add visually imperceptible perturbation to the artwork to change its underlying styles (perturbation-based methods) or embed post-training detectable watermarks in the artwork (watermark-based methods). However, when the artwork or the model has been published online, *i.e.*, modification to the original artwork or model retraining is not feasible, these strategies might not be viable.

To this end, we propose a novel method for data-use auditing in the text-to-image generation model. The general idea of ArtistAuditor is to identify if a suspicious model has been fine-tuned using specific artists' artworks by analyzing style-related features. Concretely, ArtistAuditor employs a style extractor to obtain the multi-granularity style representations and treats artworks as samplings of an artist's style. Then, ArtistAuditor queries a trained discriminator to gain the auditing decisions. The experimental results on six combinations of models and datasets show that ArtistAuditor can achieve high AUC values (> 0.937). By studying ArtistAuditor's transferability and core modules, we provide valuable insights into the practical implementation. Finally, we demonstrate the effectiveness of ArtistAuditor in real-world cases by an online platform Scenario.[1] ArtistAuditor is open-sourced at https://anonymous.4open.science/r/ArtistAuditor.

1 Introduction

Text-to-image models represent a groundbreaking advancement in generative artificial intelligence (GAI), such as DALL-E [43], Stable Diffusion [45], and Midjourney [21], which can generate realistic images from textual descriptions. These models typically function by gradually refining a random pattern of pixels into a coherent image that matches the text, making them suitable for a variety of creative and practical applications [4, 28, 31, 32, 37, 41, 48, 60].

Relevance to the Web and to the Security and Privacy Track. These models are rapidly gaining popularity among users through web platforms due to their impressive capabilities, including open API interfaces and open-source implementations. For instance, Midjourney receives around 32 million pageviews per day at around 7.5 pageviews per visit [18]. The downloads of the latest Stable Diffusion exceed 0.4 million per month. With the rapid development of text-to-image models, a user with little painting experience can use prompts to generate artwork at a professional level. As one of the sensational events, Jason M. Allen created his digital artwork with Midjourney and took first place in the digital category at the Colorado State Fair [46]. Recently, many platforms allow users to upload artworks and train the models that can generate artworks of similar style [7, 35, 48]. The ease of generating artwork using GAI might devalue the skill and expression involved in human-made artwork, diminishing the appreciation of human creativity. For instance, the artists feel that their unique styles are being appropriated when the market is flooded with AI-mimicked artworks [51]. This raises questions about dataset infringement, highly relevant to "security and privacy of machine learning and AI applications."

Table 1: Overview of the existing methods for data copyright protection. 'Tech.' refers to the core technology used by the method. 'DA' (Data Access) refers to whether the method needs access to the image or both the image and the corresponding prompt. 'DF' (Data Fidelity) stands for whether the method maintains data fidelity or not. 'TD' (Training Data) refers to whether the method needs access to the training data of the suspicious model. 'SM' (Shadow Model) refers to whether the method requires training shadow models.

Method	Goal	Tech.	DA	DF	TD	SM
[51]	Preventing misuse	Adversarial perturbation	Image	×	×	×
[58]			Image	×	×	×
[34]	Detecting misuse	Backdoor-based watermark	Both	×	×	✓
[10]			Image	×	×	×
[62]			Image	×	×	×
[3]		Membership inference	Both	✓	✓	×
[61]			Both	✓	✓	×
[38]			Image	✓	×	✓
Ours			Image	✓	×	×

Existing Solutions. To protect the intellectual property (IP) of artists, a series of strategies have been proposed [5, 6, 10, 11, 33, 34, 51, 58, 63, 66]. The existing solutions can be classified into two categories by the underlying technologies, *i.e.*, the perturbation-based methods [6, 51, 58, 66] and the watermark-based methods [10, 33, 34, 67]. The perturbation-based methods introduce subtle perturbations that alter the latent representation in the diffusion process, causing models unable to generate images as expected. The watermark-based methods inject imperceptible watermarks into artworks before they are shared. The diffusion model collects and learns the watermarked artworks. Then, the artists can validate the infringements by checking if the watermarks exist in the generated images. Membership inference (MI) [3, 53] is another technique to determine whether specific data was used to train or fine-tune the diffusion model [12, 22, 38, 61]. In Table 1, we provide an overall comparison between the existing works and ArtistAuditor.

However, previous studies face several limitations. First, both the perturbation-based and the watermark-based methods need to manipulate the original images, *i.e.*, injecting perturbation or watermark, thus compromising data fidelity. The perturbation may also diminish the model's generation quality. Second, perturbation-based and watermark-based strategies require retraining the model

[1] https://www.scenario.com/

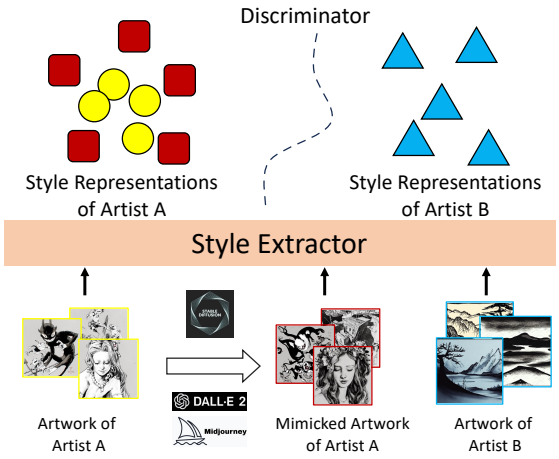

Figure 1: Intuitive explanation of ArtistAuditor**. Figures with yellow borders represent artist A's artworks, red borders indicate mimicked artworks after fine-tuning, and blue borders show B's artworks. After style extraction, red and yellow images cluster together, distant from the blue images in the latent space. The discriminator identifies the red images as imitations of the yellow ones based on feature distributions.**

to be effective. Thus, they may not suit the model already posted online. For the MI methods, the existing approaches [12, 13, 20, 25, 36, 39] for diffusion models usually require the access to structure or weights of the model, which limits their applicability in black-box auditing scenarios. Although some MI strategies target the black-box settings [22, 38, 61, 64], they are not well suited to our auditing task. We will go depth in Section 4.4 and compare them with ArtistAuditor in Section 5.

Our Proposal. In this paper, we propose a novel artwork copyright auditing method for the text-to-image models, called ArtistAuditor, which can identify data-use infringement without sacrificing the artwork's fidelity. We are inspired by the fact that artworks within an artist's style share some commonality in latent space. Thus, the auditor can mine the style-related features in an artist's works to form the auditing basis. Figure 1 provides a schematic diagram of ArtistAuditor, where the core components are the style extractor and discriminator. Since the entire feature space retains a variety of information about the artwork (*e.g.*, objects, locations, and color), the auditor needs to extract the style-related features at different levels of granularity. Then, the auditor adopts a discriminator to predict the conference score. The discriminator outputs a positive result if the generated images closely match the style of the artist; otherwise, it outputs a negative prediction. Finally, we leverage two strategies to process the confidence scores and derive the decision.

Evaluation. Our experimental results on three popular diffusion models (Stable Diffusion v2.1 [55], Stable Diffusion XL [40], and Kandinsky [44]) and two artistic datasets (Wikiart [57] and self-collected dataset) consistently achieve AUC values of ArtistAuditor above 0.937. By comparing original artworks with mimicked ones, we find that ArtistAuditor can accurately identify imitations that differ in content from the originals but pirate the artist's style. We further evaluate four influential factors from two aspects for the practical adoption of ArtistAuditor. The first aspect focuses on the transferability of ArtistAuditor. In practice, the auditor is not aware of the selected artworks or the image captioning model used to fine-tune the suspicious model. Thus, we assess the dataset and the model transferability of ArtistAuditor. When the selected artworks are disjoint with those to fine-tune the suspicious model, the auditing accuracy of ArtistAuditor only dropped by 2.6% compared to the complete overlap scenario on the Kandinsky model. For different captioning models, ArtistAuditor can still maintain an accuracy of 85.3% and a false positive rate below 13.3%. The second aspect focuses on the core modules of ArtistAuditor, namely data augmentation and distortion calibration. Data augmentation aims to increase the number of artworks available for training discriminators. Distortion calibration is used to mitigate the negative impact on auditing accuracy of potential stylistic distortions in the generation process. The results demonstrate that both modules enhance the accuracy of ArtistAuditor in most experimental settings. Finally, we show the effectiveness of ArtistAuditor in real-world cases by a commercial platform Scenario.

Contributions. Our contributions are three-fold:

- To our knowledge, ArtistAuditor is the first dataset auditing method to use multi-granularity style representations as an intrinsic fingerprint of the artist. ArtistAuditor is also an efficient solution that allows the artist to perform the auditing on consumer-grade GPU.
- We show the effectiveness of ArtistAuditor on three mainstream diffusion models. By systematically evaluating ArtistAuditor from several aspects, *i.e.*, the dataset transferability, the model transferability, and the impact of the different modules, we summarize some useful guidelines for adopting ArtistAuditor in practice.
- By implementing ArtistAuditor on the online model fine-tuning platform Scenario, we show that ArtistAuditor can serve as a potent auditing solution in real-world text-to-image scenarios.

1.1 Ethical Use of Data and Informed Consent

We strictly followed ethical guidelines by using publicly available, open-source datasets and models under licenses permitting research and educational use. As these datasets were curated and released by third parties, direct informed consent was not applicable. However, we are committed to ethical data use and will comply with all licensing terms for any future modifications or redistribution.

2 Background

2.1 Text-to-Image Generation

Generative adversarial network (GAN) [9, 16, 23] and diffusion model (DM) [21, 43, 45] have been used in text-to-image tasks. GAN in this space might struggle with the fidelity and diversity of the images. Inspired by the physical process of diffusion, where particles spread over time, DM represents a significant development in generative models. These models function through a two-phase process: a forward process that gradually adds noise to an image over a series of steps until it becomes random noise and a reverse process where the model learns to reverse this, reconstructing the image from noise. The forward process gradually adds noise to an image x_0 over a series of steps T. This process can be represented

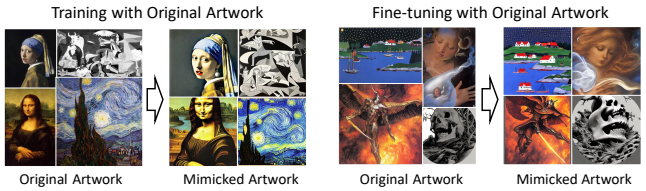

Figure 2: An example of stylistic imitation by Stable Diffusion. Left: original artwork. Right: generated artwork.

as a Markov chain where each step adds Gaussian noise.

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1-\alpha_t}\epsilon_t, \tag{1}$$

where x_t is the noisy image at step t, x_{t-1} is the image from the previous step, ϵ_t is the noise added at step t sampled from a normal distribution, *i.e.*, $\epsilon_t \sim \mathcal{N}(0, I)$. α_t is a variance schedule determining how much noise to add at each step. It's a predefined sequence of numbers between 0 and 1.

The model learns to generate images by reversing the noise addition in the reverse process. At step t, the model predicts the noise ϵ_t added in the forward process and then uses this to compute the previous step's image x_{t-1}.

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right), \tag{2}$$

where $\epsilon_\theta(x_t, t)$ is the noise predicted by the model (parameterized by θ), given x_t and the time step t. $\bar{\alpha}_t$ is the cumulative product of α_i up to step t, *i.e.*, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. The model starts with a sample of pure noise $x_T \sim \mathcal{N}(0, I)$ and applies this denoising step iteratively to arrive at a generated data point x_0. The model's training involves learning the parameters θ to predict the noise ϵ_t at each step accurately. Diffusion models excel in generating highly detailed and coherent images, showing great flexibility and stability in training, making them less prone to issues like mode collapse.

2.2 Style Piracy

Technique. The concept of style piracy in the text-to-image field refers to using diffusion models to create images that closely resemble a specific artistic style. The first way is to train the diffusion models from scratch on a large dataset of images that includes the target artist's artworks. It allows the model to learn and replicate the artist's style. A simple style piracy directly queries a text-to-image model using the artist's name. For instance, in the left of Figure 2, we utilize Stable Diffusion to imitate the style of artworks.

However, since the huge overhead for training the diffusion models, the adversary tends to fine-tune diffusion models for style piracy, *i.e.*, adjusting the diffusion models by a small set of the target artist's artwork [14, 19, 27, 47]. This dataset encompasses unique elements like specific brushwork, color schemes, and compositional techniques characteristic of the artist's style. The fine-tuning process involves continuous learning and adjustment to enhance the model's ability to apply these style characteristics accurately to various contents. In the right of Figure 2, we demonstrate the model's imitation ability after fine-tuning.

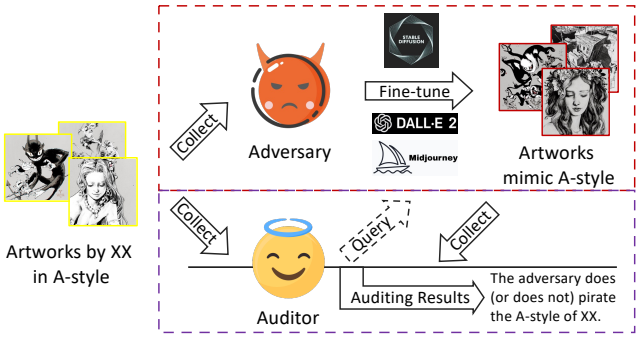

Figure 3: An example of the application scenario. The artist acquires the auditing results by comparing the style representations between the original and generated artwork.

3 Problem Statement

3.1 System and Threat Model

Application Scenarios. Comparing training the diffusion models from scratch, the adversary can easily implement the style piracy on a low-end consumer GPU by fine-tuning the models. Thus, we mainly consider the fine-tuning scenarios in this work, where the adversary collects a small set of artworks from an artist and adjusts the models' parameters to mimic the artist's style. Figure 3 illustrates a typical application case. Since many artists post their works online, adversaries can easily collect them by searching the artist's name. They fine-tune the diffusion model to generate artwork miming the artist's style. The artist stumbles upon the model's ability to generate artwork similar to his/her style and thus suspects the model's unauthorized use of his/her artwork for fine-tuning. The artist adopts ArtistAuditor to audit the suspicious model.

Auditor's Background Knowledge and Capability. For the above application scenarios, we consider the auditor to have black-box access to the suspicious text-to-image model. During the auditing, the auditor can access the artist's artworks and use a low-end consumer GPU to extract the style representations. Additionally, the auditor does not have prior knowledge of the selected artworks by the adversary. Note that this is the most general and challenging scenario for the auditor. The auditor can collect the generated images by querying the suspicious model with legitimate prompts.

3.2 Design Challenges

From the above analysis, we face two challenges during the design of the artwork copyright auditing method for text-to-image models. The primary obstacle lies in the absence of a mathematical framework to precisely define and quantify "artistic styles". Generally, the style of an artwork is defined by a multifaceted combination of elements, each contributing to its unique aesthetic and thematic identity. For instance, Claude Monet is regarded as the quintessential impressionist. Monet's work is characterized by his fascination with light and its effects on the natural world. Edgar Degas is also considered an impressionist, and his style differs significantly from that of Monet.

The second challenge is that the diffusion models often are fine-tuned with a set of artworks from multiple artists. This causes the features of these artists' artworks to interact, interfering with the effectiveness of auditing for a specific artist. Thus, the proposed method must effectively extract the unique features of an artist's artworks from the generated content to make accurate judgments.

4 Methodology

4.1 Intuition

Inspired by [15, 65], we leverage latent representations at different layers from the convolutional neural networks (CNNs) as the fingerprint of the artist's style. In CNNs, the initial layers typically capture low-level features such as edges, colors, and textures, *i.e.*, more closely related to the concrete elements of artworks. The deeper layers capture higher-level features, which represent more abstract information, like object parts or complex shapes. Then, we resort to a regression model to compress these style representations into a set of confidence scores to make the final auditing decision.

4.2 Workflow of ArtistAuditor

For clarity, an artist whose artworks are being audited is called *target artist*. If the suspicious model is fine-tuned on the target artist's artwork, the discriminator should output a positive auditing result for it; otherwise, a negative auditing result. Figure 4 illustrates the workflow of ArtistAuditor.

Step 1: Dataset Preparation (DP). The first step collects three types of artworks, *i.e.*, public artworks, generated artworks, and augmented artworks. The public artworks are the world-famous images published online, which are commonly included in the pre-training of the diffusion model [45, 49], such as the paintings of Picasso and Da Vinci. Based on these public artworks, the auditor can create a set of prompts to query the suspicious model and obtain their mimicked version. Specifically, we adopt the CLIP interrogator [1] to generate the caption for each public artwork. Then, we take these captions as prompts to query the suspicious model and get the mimicked artworks of these world-famous artists. Since the artwork of the target artist may be insufficient to train the discriminator, we utilize data augmentation to expand the target artist's works and gain the augmented artwork. We adopt the popularly used random cropping, random horizontal flipping, random cutouts, Gaussian noise [8], impulse noise [24], and color jittering [26], in existing works [17, 26, 56].

Step 2: Discriminator Construction (DC). After the first step, the auditor has public artworks, generated artworks, and augmented artworks to train a discriminator. For ease of reading, we denote the above three kinds of artwork as X_p, X_g, and X_a respectively. Recalling the design challenges in Section 3.2, we leverage a VGG model as the style extractor Φ and select the outputs of the four evenly spaced layers as the style representations. Then, for each artwork, we concatenate the style representations to form the training sample $\Phi(x)$. We use 1.0 and -1.0 as the target y, where $y = 1.0$ represents the artwork that originates from the target artist ($y = -1.0$ if it does not), to build the training set for the discriminator. Then, the loss function can be formulated as $(y - f_\theta(\Phi(x)))^2$. Since the diffusion model has distortion when imitating the artistic style, *i.e.*, there is a deviation between the original image and the generated image even under the same prompts. This distortion will cause the discriminator to mistakenly judge positive samples as negative. Thus, we integrate the distortion in the discriminator's training by measuring the difference between the public artwork and its mimicked version, *i.e.*, $\left(f_\theta(\Phi(x_g)) - f_\theta(\Phi(x_p))\right)^2$. Thus, we optimize the weights of f_θ using the following loss function.

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{dis}}, \\ \mathcal{L}_{\text{reg}} &= \left(y - f_\theta(\Phi(x))\right)^2, \\ \mathcal{L}_{\text{dis}} &= \left(f_\theta(\Phi(x_g)) - f_\theta(\Phi(x_p))\right)^2, \end{aligned} \tag{3}$$

where $\mathcal{L}_{\text{reg}}$ guides the discriminator to distinguish between the target artist's and other artists' artworks (*i.e.*, $x \in \{X_p, X_a\}$), and the distortion loss $\mathcal{L}_{\text{dis}}$ to calibrate the distortion between the generated artworks and the corresponding original artworks (*i.e.*, $x_g \in X_G, x_p \in X_p$).

Step 3: Auditing Process (AP). The auditor conducts the auditing process based on the trained discriminator. We use the same CLIP interrogator as Step 1 to create a set of captions. To encourage the suspicious model to incorporate more features of the target artists in the generated artwork, we include the target artists' information in the captions. The auditor employs the style extractor to process the generated artworks and obtain their style representations. Then, the discriminator predicts the confidence scores based on the style representations. Finally, we propose threshold-based and hypothesis-testing-based auditing mechanisms to make the auditing decision. The auditing mechanisms are detailed in Section 4.3.

4.3 Details of the Auditing Process

During the auditing process, the discriminator predicts the confidence score based on the multi-granularity style representations from the style extractor. To improve accuracy, the auditor can utilize several artworks to query the discriminator and aggregate the confidence scores to draw the decision.

A baseline strategy is to compare the average value of the confidence scores with the preset threshold. Since the discriminator is a regression model with output ranging from -1 to 1, the default threshold is set to 0. That is, if the confidence score of an artwork is higher than 0, the auditor will conclude the infringement; otherwise, there is no infringement.

The other way is to conduct hypothesis testing with the collected confidence scores. Considering the confidence scores are continuous, we select the one-sided t-test for hypothesis testing, which is used to determine if the mean of confidence scores is significantly greater than zero.

$$H_0 : \mu \leq 0, \text{ The mean value } (\mu) \text{ is equal to or less than 0.}$$

$$H_1 : \mu > 0, \text{ The mean value } (\mu) \text{ is greater than 0.}$$

For a set of confidence scores $\{c_i \mid i = 1, 2, \ldots, n\}$, t-test performs the following procedures.

1) Calculating $t = \frac{\bar{c}-0}{s/\sqrt{n}}$, where $\bar{c}$ is the average value of the samples, s is the standard deviation of the samples, and n is the number of the samples.
2) Setting the critical t-value based on the required confidence level (default 95%).
3) If the calculated t-statistic is greater than the critical t-value, the auditor will reject the null hypothesis, indicating that there is statistically significant evidence that the mean is greater than 0.

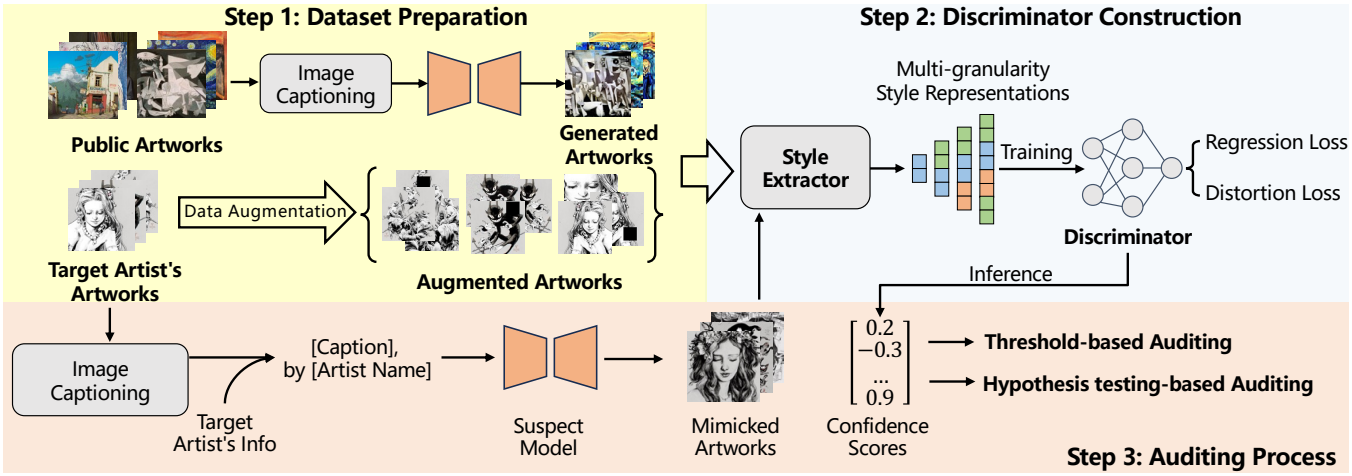

Figure 4: The workflow of ArtistAuditor **contains three steps, *i.e.*, dataset preparation, discriminator construction, and auditing process.** ArtistAuditor **first collects the public artwork and generated artwork by the suspicious model, then extracts the multi-granularity style representation to train the discriminator. Finally,** ArtistAuditor **extracts the style features of mimicked artwork and makes the auditing decision based on the outputs of the discriminator.**

4.4 Discussion

Recent works [12, 22, 38, 61] study MI methods against diffusion models. These methods can be adapted to solve the data-use auditing task. Among these, the strategies [38, 61], which are designed for black-box settings, are notable for their state-of-the-art performance. However, ArtistAuditor differs from these strategies in several essential aspects. It is worth noting that these differences are mainly due to the fact that they are optimized for different inference objectives. That is, [38] is for individual samples in the fine-tuning dataset. [61] works for the concrete property among the training samples. ArtistAuditor is optimized for the abstract property, *i.e.*, artist's style.

- **Feature Extraction.** The artwork's style is typically defined by a complex blend of elements, including low-level brushstrokes and high-level painterly motifs. Compared to [38], ArtistAuditor makes the final judgment by concatenating the features of different layers, thus better portraying the artist's artistic style.
- **Similarity Measurement.** [61] derives inference results by calculating the cosine similarity between anchor images and generated images, which is appropriate for dealing with concrete property in an image. However, artistic style is a more abstract concept. For instance, despite having completely different subjects, "Wheatfield with Crows" and "The Starry Night" both belong to the same painter, Van Gogh. Thus, we leverage an MLP model to portray the similarity of styles and derive auditing results based on the confidence scores of multiple artworks.
- **Distortion Calibration.** Due to the limitations of the model capability and the influence of other artists' artworks in the pre-training dataset, the generated artworks inevitably suffer artistic distortions. Compared to [38, 61], ArtistAuditor considers this distortion, reducing the omission of potential infringements.

5 Evaluation

We first validate the effectiveness of ArtistAuditor on three diffusion models, *i.e.*, Stable Diffusion v2.1 (SD-V2) [55], Stable Diffusion XL (SDXL) [40], and Kandinsky [44] in Section 5.2. In Section 5.3, we evaluate the dataset and the model transferability of ArtistAuditor to show that ArtistAuditor is still effective when the auditor's image captioning model (or selected artworks) differs from that (or those) of the suspicious model. After that, we further evaluate two core modules on ArtistAuditor, *i.e.*, the distortion calibration and the data augmentation in Appendix B. Finally, in Section 5.4, we utilize ArtistAuditor to audit the text-to-image models fine-tuned on a public platform Scenario.

5.1 Experimental Setup

Target Models. We adopt three text-to-image models, *Stable Diffusion v2.1 (SD-V2) [55]*, *Stable Diffusion XL (SDXL) [40]*, and *Kandinsky [44]*, which are popularly used in the prior works [33, 51, 52].

- *Stable Diffusion v2.1 (SD-V2) [55]*: SD-V2 is a high-performing and open-source model, trained on 11.5 million images from LAION [50]. It achieves state-of-the-art performance on several benchmarks [45].
- *Stable Diffusion XL (SDXL) [40]*: SDXL represents the latest advancement in diffusion model, significantly outpacing its predecessor, SD-V2, across multiple performance benchmarks. This model boasts a substantial increase in complexity, containing over 2.6 billion parameters, a stark contrast to the 865 million parameters of SD-V2. Compared to SD-V2, SDXL introduces a refiner structure to enhance the quality of image generation.
- *Kandinsky [44]*: Kandinsky is a novel text-to-image synthesis architecture that combines image-prior models with latent diffusion techniques. An image prior model, which is separately trained, maps text embeddings to image embeddings using the

Table 2: The sources of artworks.

Artist	URL Source
Xia-e	https://huaban.com/boards/58978522
Fang Li	https://huaban.com/boards/40786095
Kelek	https://gallerix.asia/storeroom/1725860866
Norris Joe	https://gallerix.asia/storeroom/1784565901
Jun Suemi	https://gallerix.asia/storeroom/2000726542
Geirrod Van Dyke	https://www.artstation.com/geirrodvandyke
Wer	https://www.gracg.com/user/p3133PKMV3r
The remaining 23 artists	https://github.com/liaopeiyuan/artbench

CLIP model. Kandinsky also features a modified MoVQ implementation serving as the image autoencoder component.

Datasets. We use the WikiArt dataset[2] following the prior works [2, 61], and randomly select fifty artists. We also build a new dataset, called Artist-30, containing the artwork of thirty artists based on fresh-published datasets [30] and publicly licensed artworks. Table 2 shows the sources of the collected artworks. We randomly selected twenty artworks from each artist.

Metrics. We adopt four metrics, *i.e.*, accuracy, area under the curve (AUC), F1 Score, false positive rate (FPR), to evaluate the performance of ArtistAuditor, which are commonly used in prior works [5, 11].

Methods. "thold" is the threshold-based auditing strategy, and "t-test" denotes the hypothesis testing-based auditing strategy. Both methods share the modules except the decision-making strategy.

Competitors. As the discussion in Section 4.4, the MI methods [38, 61] can be modified to address the data-use auditing. [38] focuses on the sample-level inference of the fine-tuning set by the similarity of the original artwork and the generated artwork. For each original artwork, [38] utilizes a classifier to predict whether it is a member or not. We slightly modified this method to align with the requirements of artist-level data-use auditing: the ratio of original artworks classified as the member is considered as the score for that artist. We use those scores to make the auditing decision. [61] exploits the feature-level consistency between the generated data and the training data to perform inference attacks. However, [61] requires the original caption of artwork in the inference, which differs from our evaluation settings. Thus, we finally add the method of [38] as a baseline in our evaluations.

Default Experimental Settings. In the evaluation, we use the following experimental settings as the default if there is no additional statement. We randomly split the artists into two groups and utilized the artworks created by the first group to fine-tune the diffusion model. For ease of reading, we note the first group of artworks as D^+ and the second group of artworks as D^-. We use CLIP interrogator [1] to generate a description for each artwork and include the artist's name in the caption, following the previous work [51]. We fine-tune the target model using dataset D^+. During the training of each artist's discriminator, we use the original artworks of each artist as positive samples and further divide them into training samples and validation samples at a ratio of 8:2. For negative samples, we randomly sample from the other artists' artwork while maintaining a positive-to-negative ratio of 1:1. In the auditing process, the threshold is set to zero.

- *The Details Settings of Fine-tuning*: Following the previous work [2], we use the corresponding fine-tuning scripts released with the models [59]. More specifically, SD-V2 is fine-tuned for 100 epochs on the dataset D^+ using the AdamW optimizer with a learning rate of 5×10^{-6}. SDXL is fine-tuned for 100 epochs on the dataset D^+ using the AdamW optimizer with a learning rate of 1×10^{-4}. As for Kandinsky, both the prior and decoder are fine-tuned for 100 epochs on the dataset D^+ using the AdamW optimizer with a learning rate of 1×10^{-4}.
- *The Details Settings of Discriminator*: We optimize the discriminator by Adam optimizer with a learning rate of 5×10^{-5}. The entire training takes 100 epochs, and we utilized an early stopping method with a patience of 10.

5.2 Overall Auditing Performance

We assess the auditing effectiveness of ArtistAuditor and its competitor [38] for SD-V2, SDXL, and Kandinsky.

Setup. We collect 20 prompts for each artist and query the target model to obtain 20 generated images. Then, the auditor puts the images into the style extractor, converts them into style representations, and gets the corresponding confidence scores based on the discriminator. Finally, we combine the auditing results of 20 artists to calculate the accuracy, AUC, F1 score, and FPR values. The experimental results are in Table 3, where the values of mean and standard variation are calculated by repeating the experiment 5 times with five random seeds $\{1, 2, 3, 4, 5\}$.

Observations. We have the following observations from Table 3. 1) ArtistAuditor archives consistent high auditing performance. The values of accuracy are higher than 0.852 for all models. These results indicate that the ArtistAuditor is highly effective at identifying unauthorized use of artist's artworks for different diffusion models. In addition, the AUC values are nearly perfect for all models, *i.e.*, more than 0.937. 2) The AUC of ArtistAuditor fluctuates on different combinations of models and datasets. ArtistAuditor achieves a remarkable AUC on Artist-30 (AUC = 1), while ArtistAuditor obtains a lower AUC of 0.937 on WikiArt. We speculate the reason is that SDXL's pre-training process uses a part of the internal dataset, which may overlap with the artworks in WikiArt. When using the same fine-tuning dataset, the AUC values of ArtistAuditor also vary on different models, such as SDXL and Kandinsky. Compared with SD-V2 and SDXL, Kandinsky switches to CLIP-ViT-G as the image encoder, significantly increasing the model's capability to generate more aesthetic pictures. 3) The FPR vaules of "t-test" usually lower than those of "thold". The selection of the threshold is an empirical process, and the average confidence score is easily misled by the outlier. Compared to "thold", "t-test" calculates the statistic t, where the number and the variance of confidence scores are also considered in the hypothesis testing. 4) ArtistAuditor is superior to the competitor in most experimental settings. The accuracy values of ArtistAuditor usually are higher than those of [38] with a lower FRP. The reason is that [38] aims at the features of the individual samples in the fine-tuning set, ignoring commonality in style between artworks by the same artist. Due to that, [38] cannot deal with the situation where the artworks used for fine-tuning

[2]https://www.wikiart.org/

Table 3: Overall auditing performance for four evaluation metrics. We report the mean and standard variance of five repeated experiments. "thold" is the threshold-based auditing strategy. "t-test" denotes the hypothesis testing-based auditing strategy.

Dataset	Model	SD-V2			SDXL			Kandinsky		
	Method / Metric	Pang et al. [38]	thold	t-test	Pang et al. [38]	thold	t-test	Pang et al. [38]	thold	t-test
WikiArt	Accuracy	0.733±0.019	**0.908±0.020**	0.896±0.015	0.813±0.009	0.852±0.010	**0.868±0.010**	0.793±0.025	**0.892±0.020**	0.852±0.010
	AUC	0.838±0.022	**0.967±0.007**	/	0.885±0.013	**0.937±0.003**	/	**1.000±0.000**	0.973±0.004	/
	F1 Score	0.661±0.027	**0.915±0.018**	0.895±0.015	0.803±0.028	0.866±0.008	**0.875±0.008**	0.802±0.006	**0.888±0.020**	0.826±0.014
	FPR	0.107±0.019	0.176±0.041	**0.096±0.032**	0.293±0.050	0.256±0.020	**0.184±0.020**	0.493±0.019	0.072±0.030	**0.000±0.000**
Artist-30	Accuracy	0.767±0.027	**0.953±0.045**	0.880±0.045	0.800±0.027	**0.947±0.016**	0.867±0.021	0.922±0.016	0.933±0.021	**0.973±0.025**
	AUC	0.986±0.004	**0.992±0.009**	/	0.923±0.030	**1.000±0.000**	/	**1.000±0.000**	0.998±0.004	/
	F1 Score	0.694±0.046	**0.951±0.049**	0.864±0.054	0.749±0.043	**0.943±0.018**	0.845±0.028	0.909±0.000	0.938±0.019	**0.975±0.023**
	FPR	**0.000±0.000**	0.027±0.033	0.013±0.027	0.000±0.000	**0.000±0.000**	0.000±0.000	0.200±0.000	0.133±0.042	**0.053±0.050**

the suspicious model are inconsistent with the artworks used for auditing, which can be further corroborated by the results in the dataset transferability of Section 5.3.

5.3 Transferability of ArtistAuditor

The auditor is not aware of the selected artworks or the image captioning model used to fine-tune the suspicious model. Therefore, this section aims to assess the transferability of ArtistAuditor. We begin by evaluating the dataset transferability when the artworks used for auditing differ from those used to fine-tune the suspicious model. Next, we assess model transferability when the auditor's image captioning model differs from that of the suspicious model.

Dataset Transferability. We consider two scenarios, *i.e.*, the partial overlap and the disjoint cases. In the partial overlap scenario, the artworks used by the suspicious model are half overlapped with the artworks used by the auditor. In the disjoint scenario, the auditor has a set of artworks from the target artist, and these artworks are different from the artworks used in the fine-tuning of the suspicious model. For each experimental setting, we conduct five replicate experiments with random seeds set to {1, 2, 3, 4, 5}, and report the mean and variance of the results.

Table 4 shows the effectiveness of ArtistAuditor in auditing artistic style piracy across different degrees of dataset overlaps. 1) When the artworks partially overlap, the performance of ArtistAuditor slightly decreases. ArtistAuditor still remains effective with AUC > 0.964 and FPR < 0.133. For example, for the SDXL model, ArtistAuditor achieves an auditing accuracy of up to 0.920, which is only 0.027 lower than that of the complete overlap scenario. This indicates the internal consistency of the artist's work style, which can be extracted by the style extractor and used as an auditing basis for whether infringement of the artwork occurs.

2) The most significant performance drop is observed in the disjoint scenario, particularly in accuracy and F1 scores. Compared to [38], ArtistAuditor can still effectively detect the mimicked artworks. Especially on the Kandinsky model, the auditing accuracy of ArtistAuditor only dropped by 0.026 compared to the complete overlap scenario. The comparison demonstrates that ArtistAuditor does not rely on the overfitting of individual artwork but rather learns to discriminate based on the internal commonality of the artist's style.

Model Transferability. The suspicious model may apply a different captioning model from that of the auditor to generate prompts, leading to the distribution of prompts being different. Figure 5 compare the captions generated by two different image captioning models, *i.e.*, CLIP [42] and BLIP [29]. For each experimental setting, we conduct five replicate experiments with random seeds set to {1, 2, 3, 4, 5}, and report the mean and variance of the results.

Table 4: Dataset Transferability of ArtistAuditor. "thold" indicates the threshold-based auditing strategy. "t-test" denotes the hypothesis testing-based auditing strategy. "Partially" and "Disjoint" refer to the dataset's partial overlap and disjoint scenarios. Table 7 shows more details.

Model	Setting	Partially			Disjoint		
	Metric	[38]	thold	t-test	[38]	thold	t-test
SD-V2	Accuracy	0.789	**0.800**	0.760	0.556	**0.727**	0.687
	AUC	**0.991**	0.964	/	0.699	**0.956**	/
	F1 Score	0.745	**0.754**	0.683	0.281	**0.623**	0.543
	FPR	0.000	0.000	0.000	0.000	0.000	0.000
SDXL	Accuracy	0.689	**0.920**	0.873	0.511	**0.727**	0.633
	AUC	0.921	**1.000**	/	0.872	**0.980**	/
	F1 Score	0.576	**0.912**	0.855	0.148	**0.622**	0.419
	FPR	0.000	0.000	0.000	0.000	0.000	0.000
Kandinsky	Accuracy	0.933	0.933	**0.967**	0.711	**0.907**	0.853
	AUC	0.936	**0.996**	/	0.744	**0.982**	/
	F1 Score	0.923	0.938	**0.967**	0.667	**0.896**	0.826
	FPR	0.187	0.133	**0.053**	0.190	0.013	**0.000**

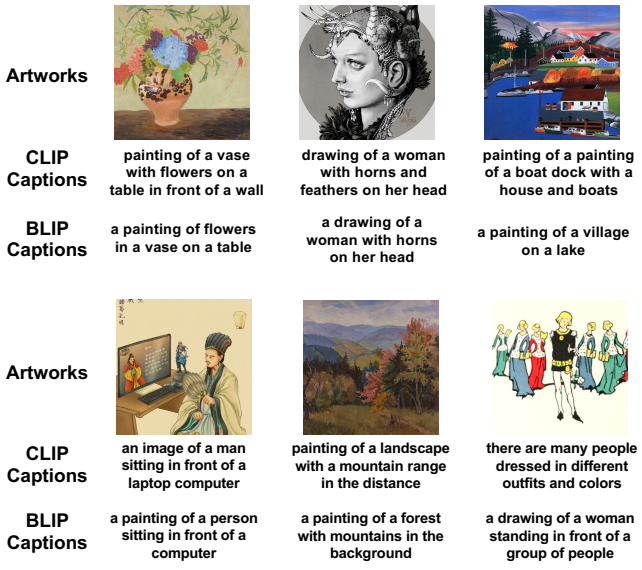

Figure 5: Some artworks with their captions generated by CLIP and BLIP, respectively.

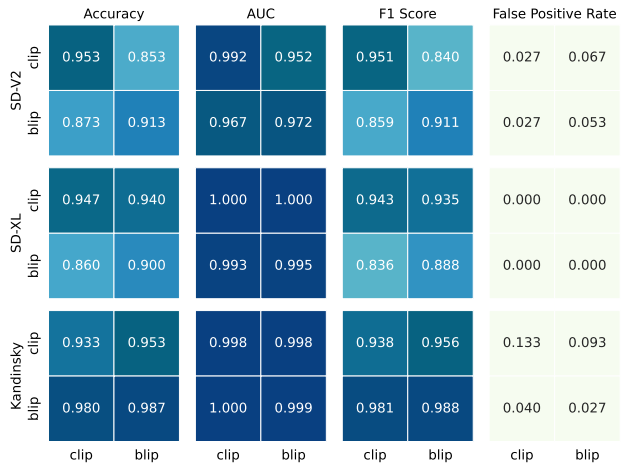

Figure 6: Model Transferability of ArtistAuditor**. The x-axis is the image captioning model used in suspicious models. The y-axis is the image captioning model used by the auditor. Table 8 shows more details.**

Figure 6 shows the model transferability of ArtistAuditor. 1) When the same image captioning model is used by both the suspicious model and the auditor, ArtistAuditor achieves high auditing performance. For instance, ArtistAuditor performs an auditing accuracy of 0.947 on the SDXL model with an FPR equal to 0. Kandinsky has a higher FPR (0.133) but maintains reasonable accuracy (0.933) and F1 scores (0.938). 2) The results show a slight decrease in auditing performance when different image captioning models are used. This is particularly evident in the SD-V2 model, where the accuracy drops from 0.953 to 0.853, and the F1 score drops from 0.951 to 0.840. However, the AUC values remain high, indicating strong discriminative power despite the variation in prompt generation. On one hand, when the artwork's content is fixed, the distribution of suitable captions is limited. On the other hand, ArtistAuditor mainly grasps the stylistic characteristics of the artist rather than fitting specific artwork, making it robust to the caption's changes.

5.4 Real-World Performance

We demonstrate the effectiveness of ArtistAuditor in real-world applications by an online model fine-tuning platform Scenario. After the user uploads a set of artworks, the platform fine-tunes a model to mimic the artistic style and returns an API for the user to generate mimicked artworks.

Setup. Recalling Section 5.3, the auditor is not aware of the specific artworks used to fine-tune the suspicious model. Thus, aligning with Table 4, we provide the auditing performance in complete overlap, partial overlap, and disjoint cases. Due to the limited number of images for single fine-tuning on Scenario, we randomly pick 10 artworks from each artist and upload them to fine-tune the model. We perform auditing for three different artists separately.

Observations. We have the following observations from Table 5. 1) ArtistAuditor achieves correct auditing results under all experimental settings. The auditing results of ArtistAuditor on three artists are significantly greater than the threshold 0, meaning that ArtistAuditor is a valid auditing solution. 2) ArtistAuditor maintains high auditing performance under dataset transfer settings. Compared to the auditing results in Section 5.3, ArtistAuditor seems to show better dataset transferability on the online platform. The reason is mainly that online platforms have better computing power, which makes it possible to get a good artistic imitation even in a disjoint case (please refer to Figure 8 for the generated images).

Table 5: The average of confidence scores predicted by ArtistAuditor**. The results are significantly higher than 0, meaning that** ArtistAuditor **is valid for real-world auditing.**

Confidence Score / Setting / Artist	Completely	Partially	Disjoint
Dela Rosa	0.840	0.874	0.891
Xia-e	0.380	0.437	0.501
David Michael Hinnebusch	0.745	0.762	0.807

6 Discussion

Highlights of ArtistAuditor**.** 1) ArtistAuditor is the first data-use auditing method for the diffusion model without the requirement of the model's retraining or modification to original artworks. 2) By comprehensively evaluating ArtistAuditor under different experimental settings, such as the dataset transferability, the model transferability, the data augmentation, and the distortion calibration, we conclude some useful observations for adopting ArtistAuditor. 3) We apply ArtistAuditor to audit the fine-tuned models on an online platform. The auditing decisions are all correct, demonstrating that ArtistAuditor is an effective and efficient strategy for practical use.

Limitations and Future Work. We discuss the limitations of ArtistAuditor and promising directions for further improvements. 1) From Section 5.2, the accuracy of ArtistAuditor decreases when more artists' works are involved in the fine-tuning process. Thus, it is interesting to enhance ArtistAuditor by mining more personalized features from the artists' works. 2) Adversarial perturbation may diminish the auditing accuracy of ArtistAuditor. Thus, integrating adversarial training methods is a potential mitigation approach.

7 Conclusion

In this work, we propose a novel artwork auditing method for text-to-image models based on the insight that the multi-granularity latent representations of a CNN model can serve as the intrinsic fingerprint of an artist. Through extensive experiments, we show that ArtistAuditor is an effective and efficient solution for protecting the intellectual property of artworks. The experimental results on six combinations of models and datasets show that ArtistAuditor can achieve high AUC values (> 0.937). and the auditing process can be performed on a consumer-grade GPU. By evaluating the dataset transferability, the captioning model transferability, the impact of data augmentation, and the impact of distortion calibration, we conclude several important observations for adopting ArtistAuditor in practice. Finally, we utilize the online commercial platform Scenario to examine the practicality of ArtistAuditor, and show that ArtistAuditor behaves excellently on real-world auditing.

References

[1] clip-interrogator. https://github.com/pharmapsychotic/clip-interrogator?tab=readme-ov-file.

[2] B. Cao, C. Li, T. Wang, J. Jia, B. Li, and J. Chen. IMPRESS: Evaluating the Resilience of Imperceptible Perturbations Against Unauthorized Data Usage in Diffusion-Based Generative AI. *ArXiv*, abs/2310.19248, 2023.

[3] D. Chen, N. Yu, Y. Zhang, and M. Fritz. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *ACM CCS*, 2019.

[4] J. Chen, J. Wang, X. Ma, Y. Sun, J. Sun, P. Zhang, and P. Cheng. QuoTe: Quality-oriented Testing for Deep Learning Systems. *ACM Transactions on Software Engineering and Methodology*, 2023.

[5] M. Chen, Z. Zhang, T. Wang, M. Backes, and Y. Zhang. FACE-AUDITOR: Data Auditing in Facial Recognition Systems. In *USENIX Security*, 2023.

[6] R. Chen, H. Jin, J. Chen, and L. Sun. EditShield: Protecting Unauthorized Image Editing by Instruction-guided Diffusion Models. *ArXiv*, abs/2311.12066, 2023.

[7] CIVITAI. What the heck is Civitai? https://civitai.com/content/guides/what-is-civitai, 2022.

[8] J. M. Cohen, E. Rosenfeld, and J. Z. Kolter. Certified Adversarial Robustness via Randomized Smoothing. *ArXiv*, abs/1902.02918, 2019.

[9] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 2017.

[10] Y. Cui, J. Ren, H. Xu, P. He, H. Liu, L. Sun, and J. Tang. DiffusionShield: A Watermark for Copyright Protection against Generative Diffusion Models. *ArXiv*, abs/2306.04642, 2023.

[11] L. Du, M. Chen, M. Sun, S. Ji, P. Cheng, J. Chen, and Z. Zhang. ORL-Auditor: Dataset Auditing in Offline Deep Reinforcement Learning. In *NDSS*. Internet Society, 2024.

[12] J. Duan, F. Kong, S. Wang, X. Shi, and K. Xu. Are diffusion models vulnerable to membership inference attacks? *ArXiv*, 2023.

[13] W. Fu, H. Wang, C. Gao, G. Liu, Y. Li, and T. Jiang. A probabilistic fluctuation based membership inference attack for diffusion models. *ArXiv*, 2023.

[14] R. Gal, Y. Alaluf, Y. Atzmon, O. Patashnik, A. H. Bermano, G. Chechik, and D. Cohen-or. An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion. In *ICLR*, 2022.

[15] L. A. Gatys, A. S. Ecker, and M. Bethge. A Neural Algorithm of Artistic Style. *arXiv preprint arXiv:1508.06576*, 2015.

[16] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative Adversarial Networks. *Communications of the ACM*, 2014.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015.

[18] C. Heidorn. Mind-Boggling Midjourney Statistics in 2023. Tokenized, 2023.

[19] E. J. Hu, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*, 2021.

[20] H. Hu and J. Pang. Membership inference of diffusion models. *ArXiv*, 2023.

[21] N. Ivanenko. Midjourney v4: An incredible new version of the ai image generator, 2022.

[22] N. Kandpal, K. Pillutla, A. Oprea, P. Kairouz, C. A. Choquette-Choo, and Z. Xu. User inference attacks on large language models. *ArXiv*, 2023.

[23] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila. Alias-free generative adversarial networks. In *NeurIPS*, 2021.

[24] M. A. Koli and B. S. Literature survey on impulse noise reduction. *Signal & Image Processing : An International Journal*, 4:75–95, 2013.

[25] F. Kong, J. Duan, R. Ma, H. Shen, X. lan Zhu, X. Shi, and K. Xu. An efficient membership inference attack for the diffusion model by proximal initialization. *ArXiv*, 2023.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*, 60:84 – 90, 2012.

[27] N. Kumari, B. Zhang, R. Zhang, E. Shechtman, and J.-Y. Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, 2023.

[28] H. Li, Y. Yang, M. Chang, H. Feng, Z. hai Xu, Q. Li, and Y. ting Chen. SRDiff: Single Image Super-Resolution with Diffusion Probabilistic Models. *Neurocomputing*, 2021.

[29] J. Li, D. Li, and Others. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. In *ICML*, 2022.

[30] P. Liao, X. Li, X. Liu, and K. Keutzer. The ArtBench Dataset: Benchmarking Generative Models with Artworks. *arXiv preprint arXiv:2206.11404*, 2022.

[31] G. Liu. The world's smartest artificial intelligence just made its first magazine cover. Cosmopolitan, 2022.

[32] P. Liu, J. Liu, et al. How ChatGPT is Solving Vulnerability Management Problem. *arXiv preprint arXiv:2311.06530*, 2023.

[33] G. Luo, J. Huang, M. Zhang, Z. Qian, S. Li, and X. Zhang. Steal My Artworks for Fine-tuning? A Watermarking Framework for Detecting Art Theft Mimicry in Text-to-Image Models. *ArXiv*, abs/2311.13619, 2023.

[34] Y. Ma, Z. Zhao, X. He, Z. Li, M. Backes, and Y. Zhang. Generative Watermarking Against Unauthorized Subject-Driven Image Synthesis. *ArXiv*, abs/2306.07754, 2023.

[35] V. Madan, H. Hotz, and X. Ma. Fine-tune Text-to-image Stable Diffusion Models with Amazon SageMaker JumpStart. https://aws.amazon.com/blogs/machine-learning/fine-tune-text-to-image-stable-diffusion-models-with-amazon-sagemaker-jumpstart/i, 2023.

[36] T. Matsumoto, T. Miura, and N. Yanai. Membership inference attacks against diffusion models. *IEEE SPW*, 2023.

[37] J. Meng, Z. Yang, Z. Zhang, Y. Geng, R. Deng, P. Cheng, J. Chen, and J. Zhou. SePanner: Analyzing Semantics of Controller Variables in Industrial Control Systems based on Network Traffic. In *ACSAC*, 2023.

[38] Y. Pang and T. Wang. Black-box membership inference attacks against fine-tuned diffusion models. *ArXiv*, 2023.

[39] Y. Pang, T. Wang, X. Kang, M. Huai, and Y. Zhang. White-box membership inference attacks against diffusion models. *ArXiv*, 2023.

[40] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Muller, J. Penna, and R. Rombach. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis. *ArXiv*, abs/2307.01952, 2023.

[41] N. Popli. He Used AI to Publish a Children's Book in a Weekend. Artists Are Not Happy About It. https://time.com/6240569/ai-childrens-book-alice-and-sparkle-artists-unhappy/, 2022.

[42] A. Radford, J. W. Kim, et al. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*, 2021.

[43] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation, 2021.

[44] A. Razzhigaev, A. Shakhmatov, A. Maltseva, V. Arkhipkin, I. Pavlov, I. Ryabov, A. Kuts, A. Panchenko, A. Kuznetsov, and D. Dimitrov. Kandinsky: an Improved Text-to-Image Synthesis with Image Prior and Latent Diffusion. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

[45] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[46] K. Roose. An A.I.-Generated Picture Won an Art Prize. Artists Aren't Happy. https://www.nytimes.com/2022/09/02/technology/ai-artificial-intelligence-artists.html, 2022.

[47] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023.

[48] Scenario.gg. AI-generated Aame Assets. https://www.scenario.gg/, 2022.

[49] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, et al. Laion-5b: An Open Large-scale Dataset for Training Next Generation Image-text Models. *NeurIPS*, 2022.

[50] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models, 2022.

[51] S. Shan, J. Cryan, E. Wenger, H. Zheng, R. Hanocka, and B. Y. Zhao. GLAZE: Protecting Artists from Style Mimicry by Text-to-Image Models. In *USENIX Security*, 2023.

[52] S. Shan, W. Ding, J. Passananti, H. Zheng, and B. Y. Zhao. Prompt-Specific Poisoning Attacks on Text-to-Image Generative Models. *ArXiv*, abs/2310.13828, 2023.

[53] R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18, 2016.

[54] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein. Diffusion Art or Digital Forgery? Investigating Data Replication in Diffusion Models. *CoRR abs/2212.03860*, 2022.

[55] Stability AI. Stable Diffusion v2.1 and DreamStudio Updates 7-Dec 22, 2022. https://stability.ai/blog/stablediffusion2-1-release7-dec-2022.

[56] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2014.

[57] W. R. Tan, C. S. Chan, H. Aguirre, and K. Tanaka. Improved ArtGAN for Conditional Synthesis of Natural Image and Artwork. *IEEE Transactions on Image Processing*, 2019.

[58] T. Van Le, H. Phung, T. H. Nguyen, Q. Dao, N. N. Tran, and A. Tran. Anti-DreamBooth: Protecting Users from Personalized Text-to-image Synthesis. In *ICCV*, 2023.

[59] P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, and T. Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022.

[60] K. Wang, J. Wang, C. M. Poskitt, X. Chen, J. Sun, and P. Cheng. K-ST: A Formal Executable Semantics of the Structured Text Language for PLCs. *IEEE Transactions on Software Engineering*, 2023.

[61] L. Wang, J. Wang, J. Wan, L. Long, Z. Yang, and Z. Qin. Property existence inference against generative models. In *USENIX Security Symposium*, 2024.
[62] Z. Wang, C. Chen, L. Lyu, D. N. Metaxas, and S. Ma. Diagnosis: Detecting unauthorized data usages in text-to-image diffusion models. In *ICLR*, 2023.
[63] C. Wei, W. Meng, Z. Zhang, M. Chen, M. Zhao, W. Fang, L. Wang, Z. Zhang, and W. Chen. LMSanitator: Defending Task-agnostic Backdoors Against Prompt-tuning. In *NDSS*. Internet Society, 2024.
[64] M. Zhang, N. Yu, R. Wen, M. Backes, and Y. Zhang. Generated distributions are all you need for membership inference attacks against generative models. *IEEE/CVF WACV*, 2024.
[65] Y. Zhang, F. Tang, W. Dong, H. Huang, C. Ma, T.-Y. Lee, and C. Xu. Domain Enhanced Arbitrary Image Style Transfer via Contrastive Learning. In *ACM SIGGRAPH*, 2022.
[66] Z. Zhao, J. Duan, K. Xu, C. Wang, R. Guo, and X. Hu. Can Protective Perturbation Safeguard Personal Data from Being Exploited by Stable Diffusion? *ArXiv*, abs/2312.00084, 2023.
[67] H. Zhu, M. Liu, C. Fang, R. Deng, and P. Cheng. Detection-Performance Tradeoff for Watermarking in Industrial Control Systems. *IEEE Transactions on Information Forensics and Security*, 2023.

A Data Augmentation

This section elaborates on the data augmentation strategies used in Section 4.2.

- **Random Cropping.** It involves selecting a random portion of the image and using only that cropped part for training, which helps the model focus on different parts of the image and learn more comprehensive features.
- **Random Horizontal Flipping.** This augmentation technique flips images horizontally at random. This is particularly useful for teaching the model that the orientation of objects can vary, and it should still be able to recognize the object regardless of its mirrored position. Horizontally or vertically flipping the artwork
- **Random Cutouts.** It involves randomly removing squares or rectangles of various sizes from an image during training. This forces the model to focus on less information and learn to make predictions based on partial views of objects. It is beneficial for enhancing the model's ability to focus on the essential features of the image without overfitting to specific details.
- **Gaussian noise.** It injects noise that follows a Gaussian distribution into image pixels. This technique helps the model become more robust to variations in pixel values and can improve its ability to generalize well on new, unseen data.
- **Impulse noise.** Impulse noise, also known as salt-and-pepper noise, randomly alters the pixel values in images, turning some pixels completely white or black. Training with impulse noise can help the model learn to ignore significant but irrelevant local variations in the image data.
- **Color jittering.** It encompasses adjustments to brightness, saturation, contrast, and hue of the image randomly, which is beneficial for preparing the model to handle images under various lighting conditions and color settings.

B Ablation Study

Impact of Data Augmentation. Recalling Section 4.2, the data augmentation aims to expand the number of artworks for training discriminators. We compare the performance of ArtistAuditor with and without data augmentation.

The results in columns "w/o DA" and "Baseline" of Table 6 show that data augmentation significantly enhances auditing performance. For instance, the accuracy of ArtistAuditor increases from 0.633 to 0.947 in the SDXL model, and from 0.647 to 0.933 in the Kandinsky model. Data augmentation significantly increases the number and diversity of artworks, preventing the discriminator from overfitting to style-irrelevant features.

Table 6: Impact of data augmentation and distortion calibration. "w/o DA" shows the auditing performance without data augmentation. "w/o DC" shows the auditing performance without distortion calibration. Table 9 shows more details.

Model	Setting	w/o DA		w/o DC		Baseline	
	Metric	thold	t-test	thold	t-test	thold	t-test
SD-V2	Accuracy	0.953	0.853	0.927	0.867	0.953	0.880
	AUC	0.994	/	0.995	/	0.992	/
	F1 Score	0.951	0.825	0.920	0.845	0.951	0.864
	FPR	0.013	0.000	0.000	0.000	0.027	0.013
SDXL	Accuracy	0.953	0.893	0.633	0.620	0.947	0.867
	AUC	0.997	/	0.874	/	1.000	/
	F1 Score	0.951	0.879	0.411	0.372	0.943	0.845
	FPR	0.000	0.000	0.000	0.000	0.000	0.000
Kandinsky	Accuracy	0.880	0.913	0.647	0.620	0.933	0.973
	AUC	0.977	/	0.850	/	0.998	/
	F1 Score	0.893	0.920	0.460	0.382	0.938	0.975
	FPR	0.240	0.173	0.013	0.000	0.133	0.053

Impact of Distortion Calibration. In Section 4.2, we try to calibrate the style distortion between the artworks generated by the suspicious model and the original artworks used in its training process. The calibration dataset comprises artworks from two sources: public artworks and generated artworks. We evaluate the impact of distortion calibration.

The results in columns "w/o DC" and "Baseline" of Table 6 show that the distortion calibration generally improves accuracy for both auditing strategies. For example, the auditing accuracy of ArtistAuditor on Kandinsky increases from 0.880 to 0.933, while the FPR decreases from 0.240 to 0.133. With the help of distortion calibration, the discriminator can effectively learn the subtle differences between the style of original artworks and the style of model-generate artworks. This makes ArtistAuditor more robust in detecting unauthorized usage, ensuring better protection of IP.

C Target Models' Performance

We first investigate the stylistic imitation ability of the target model, as shown in Figure 7. The first row shows the original artworks created by artists. The second row shows generated artworks without fine-tuning the target models with the original artwork. The third row shows mimicked artworks by the target models fine-tuned on the original artworks.

By comparing these three parts in Figure 7, it becomes apparent that the target model, after being fine-tuned on the original artworks, exhibits a discernible ability to imitate artistic styles. However, detecting the imitation of certain artwork is not immediately evident, making it challenging to ascertain through direct visual inspection, such as the image in the lower left corner of Figure 7. This underscores the necessity of ArtistAuditor to identify potential infringements.

D Related Work

In this section, we go into depth about the existing solutions, as the extension of that in Section 1. As diffusion models continue

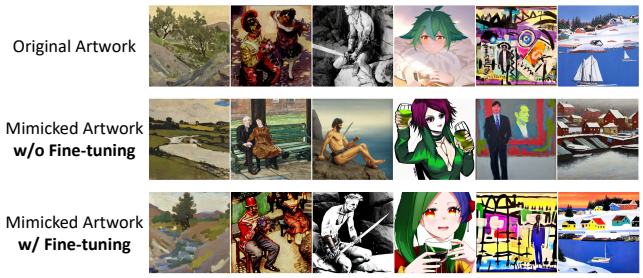

Figure 7: Target models' performance. The first row displays the original artwork created by the artists. The second row displays imitations generated by the text-to-image model before its fine-tuning on the original artwork. The final row showcases the imitations created after fine-tuning.

to evolve and gain popularity, users can now create a vast array of generative works at a low cost, which leads to the negative effects of the replication becoming more acute [54]. Especially the artist community is concerned about the copyright infringement of their work [7, 35, 48]. Recently, researchers have proposed a lot of countermeasures to solve this issue.

Perturbation-based Method. The artists can introduce slight perturbations that modify the latent representation during the diffusion process, preventing models from generating the expected images. Shan *et al.* [51] introduce Glaze, a tool that allows artists to apply "style cloaks" to their artwork, introducing subtle perturbations that mislead generative models attempting to replicate a specific artist's style. Similarly, Anti-DreamBooth [58] is a defense system designed to protect against the misuse of DreamBooth by adding slight noise perturbations to images before they are published, thereby degrading the quality of images generated by models trained on these perturbed datasets. Chen *et al.* [6] propose EditShield, a protection method that introduces imperceptible perturbations to shift the latent representation during the diffusion process, causing models to produce unrealistic images with mismatched subjects.

However, the goal of adversarial perturbation is to disrupt the learning process of diffusion models, which is orthogonal to the copyright auditing focus of this paper. Moreover, adversarial perturbation essentially blocks any legitimate use of subject-driven synthesis based on protected images.

Watermark-based Method. This framework adds subtle watermarks to digital artworks to protect copyrights while preserving the artist's expression. Cui *et al.* [10] construct the watermark by converting the copyright message into an ASCII-based binary sequence and then translating it into a quaternary sequence. During the copyright auditing, they adopt a ResNet-based decoder to recover the watermarks from the generated images by a third-party model. Luo *et al.* [33] choose to embed subtle watermarks into digital artworks to protect copyrights while preserving the artist's style. If used as training data, these watermarks become detectable markers, where the auditor can reveal unauthorized mimicry by analyzing their distribution in generated images. Ma *et al.* [34] propose GenWatermark, a novel system that jointly trains a watermark generator and detector. By integrating the subject-driven synthesis process during training, GenWatermark fine-tunes the detector with synthesized images, boosting detection accuracy and ensuring subject-specific watermark uniqueness.

However, given that digital artworks are already in the public domain, artists must utilize a post-publication mechanism that does not depend on the prior insertion of altered samples into the dataset. In contrast, watermarking constitutes a preemptive measure, necessitating the integration of manipulated samples into the dataset before its release.

Table 7: Dataset Transferability of ArtistAuditor. "thold" indicates the threshold-based auditing strategy. "t-test" denotes the hypothesis testing-based auditing strategy.

Model	Setting	Partially Overlap			Disjoint		
	Method / Metric	[38]	thold	t-test	[38]	thold	t-test
SD-V2	Accuracy	0.789±0.042	0.800±0.021	0.760±0.025	0.556±0.031	0.727±0.013	0.687±0.016
	AUC	0.991±0.007	0.964±0.008	/	0.699±0.034	0.956±0.015	/
	F1 Score	0.745±0.057	0.754±0.026	0.683±0.043	0.281±0.084	0.623±0.026	0.543±0.035
	FPR	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000
SDXL	Accuracy	0.689±0.031	0.920±0.027	0.873±0.013	0.511±0.031	0.727±0.025	0.633±0.021
	AUC	0.921±0.012	1.000±0.000	/	0.872±0.011	0.980±0.020	/
	F1 Score	0.576±0.043	0.912±0.031	0.855±0.017	0.148±0.105	0.622±0.047	0.419±0.053
	FPR	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000
Kandinsky	Accuracy	0.933±0.000	0.933±0.030	0.967±0.000	0.711±0.031	0.907±0.044	0.853±0.040
	AUC	0.936±0.022	0.996±0.004	/	0.744±0.017	0.982±0.013	/
	F1 Score	0.923±0.024	0.938±0.026	0.967±0.001	0.667±0.067	0.896±0.055	0.826±0.051
	FPR	0.187±0.070	0.133±0.060	0.053±0.027	0.190±0.067	0.013±0.027	0.000±0.000

Table 8: Model Transferability of ArtistAuditor. We use CLIP and BLIP as image captioning models. For each combination, the former is the image captioning model used by the auditor. The later is the image captioning model used in suspicious models.

Model	Image Captioning Model	CLIP+CLIP		CLIP+BLIP		BLIP+CLIP		BLIP+BLIP	
	Method / Metric	thold	t-test	thold	t-test	thold	t-test	thold	t-test
SD-V2	Accuracy	0.953±0.045	0.880±0.045	0.853±0.027	0.827±0.025	0.873±0.025	0.807±0.025	0.913±0.027	0.833±0.021
	AUC	0.992±0.009	/	0.952±0.011	/	0.967±0.007	/	0.972±0.009	/
	F1 Score	0.951±0.049	0.864±0.054	0.840±0.033	0.789±0.036	0.859±0.028	0.759±0.039	0.911±0.026	0.806±0.025
	FPR	0.027±0.033	0.013±0.027	0.067±0.000	0.000±0.000	0.027±0.033	0.000±0.000	0.053±0.050	0.027±0.033
SDXL	Accuracy	0.947±0.016	0.867±0.021	0.940±0.025	0.873±0.039	0.860±0.025	0.767±0.037	0.900±0.021	0.860±0.033
	AUC	1.000±0.000	/	1.000±0.000	/	0.993±0.004	/	0.995±0.007	/
	F1 Score	0.943±0.018	0.845±0.028	0.935±0.029	0.853±0.050	0.836±0.033	0.693±0.060	0.888±0.026	0.835±0.046
	FPR	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000
Kandinsky	Accuracy	0.933±0.021	0.973±0.025	0.953±0.016	0.967±0.021	0.980±0.027	0.980±0.016	0.987±0.027	0.973±0.013
	AUC	0.998±0.004	/	0.998±0.002	/	1.000±0.000	/	0.999±0.002	/
	F1 Score	0.938±0.019	0.975±0.023	0.956±0.015	0.966±0.021	0.981±0.025	0.979±0.017	0.988±0.025	0.973±0.014
	FPR	0.133±0.042	0.053±0.050	0.093±0.033	0.027±0.033	0.040±0.053	0.000±0.000	0.027±0.053	0.013±0.027

Table 9: Impact of data augmentation and distortion calibration. "w/o DA" shows the auditing performance without data augmentation. "w/o DC" shows the auditing performance without distortion calibration.

Model	Setting	w/o Data Augmentation		w/o Distortion Calibration		Baseline	
	Method	thold	t-test	thold	t-test	thold	t-test
SD-V2	Accuracy	0.927±0.025	0.867±0.021	0.953±0.016	0.853±0.045	0.953±0.045	0.880±0.045
	AUC	0.995±0.005	/	0.994±0.008	/	0.992±0.009	/
	F1 Score	0.920±0.029	0.845±0.028	0.951±0.018	0.825±0.060	0.951±0.049	0.864±0.054
	FPR	0.000±0.000	0.000±0.000	0.013±0.027	0.000±0.000	0.027±0.033	0.013±0.027
SDXL	Accuracy	0.633±0.052	0.620±0.062	0.953±0.016	0.893±0.033	0.947±0.016	0.867±0.021
	AUC	0.874±0.069	/	0.997±0.002	/	1.000±0.000	/
	F1 Score	0.411±0.117	0.372±0.149	0.951±0.018	0.879±0.042	0.943±0.018	0.845±0.028
	FPR	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000	0.000±0.000
Kandinsky	Accuracy	0.647±0.027	0.620±0.034	0.880±0.016	0.913±0.016	0.933±0.021	0.973±0.025
	AUC	0.850±0.085	/	0.977±0.017	/	0.998±0.004	/
	F1 Score	0.460±0.075	0.382±0.090	0.893±0.013	0.920±0.014	0.938±0.019	0.975±0.023
	FPR	0.013±0.027	0.000±0.000	0.240±0.033	0.173±0.033	0.133±0.042	0.053±0.050

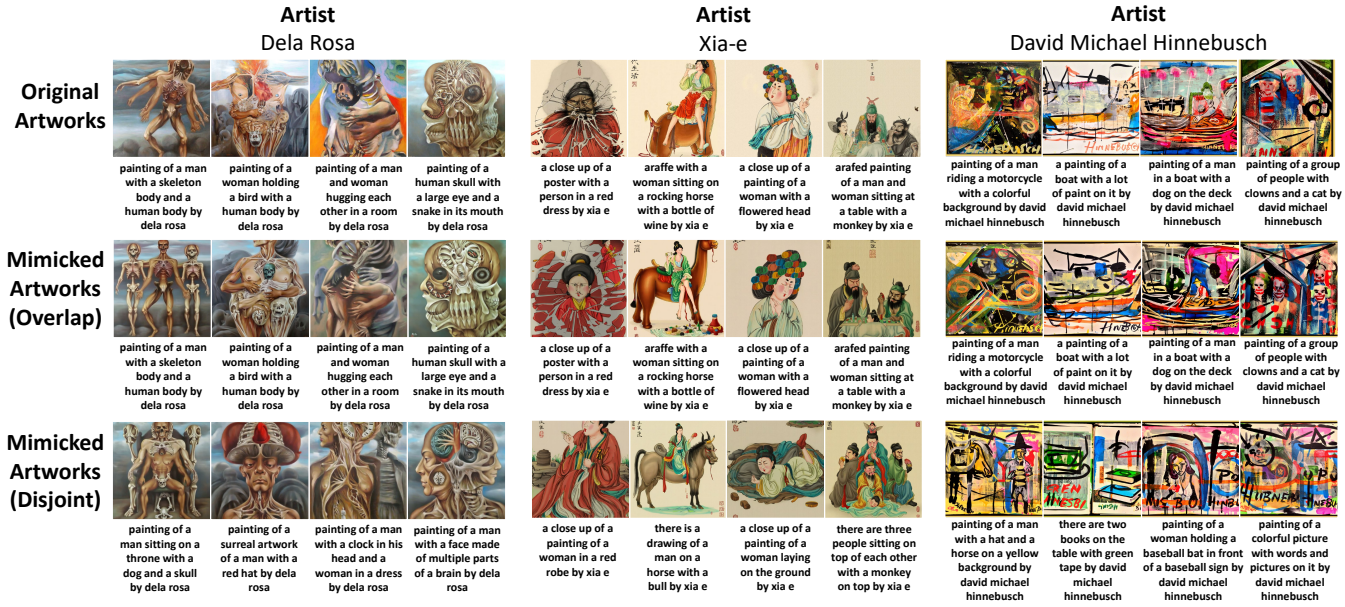

Figure 8: The original artworks and mimicked artworks of the online platform Scenario. The text below the artwork is the corresponding prompt.