ProMark: Proactive Diffusion Watermarking for Causal Attribution

Vishal Asnani[1,2] John Collomosse[1,3] Tu Bui[3] Xiaoming Liu[2] Shruti Agarwal[1]
[1]Adobe Research, [2]Michigan State University, [3]University of Surrey
{asnanivi,liuxm}@msu.edu {collomos,shragarw}@adobe.com t.v.bui@surrey.ac.uk

Abstract

Generative AI (GenAI) is transforming creative workflows through the capability to synthesize and manipulate images via high-level prompts. Yet creatives are not well supported to receive recognition or reward for the use of their content in GenAI training. To this end, we propose ProMark, a causal attribution technique to attribute a synthetically generated image to its training data concepts like objects, motifs, templates, artists, or styles. The concept information is proactively embedded into the input training images using imperceptible watermarks, and the diffusion models (unconditional or conditional) are trained to retain the corresponding watermarks in generated images. We show that we can embed as many as 2^{16} unique watermarks into the training data, and each training image can contain more than one watermark. ProMark can maintain image quality whilst outperforming correlation-based attribution. Finally, several qualitative examples are presented, providing the confidence that the presence of the watermark conveys a causative relationship between training data and synthetic images.

1. Introduction

GenAI is able to create high-fidelity synthetic images spanning diverse concepts, largely due to advances in diffusion models, *e.g.* DDPM [18], DDIM [23], LDM [28]. GenAI models, particularly diffusion models, have been shown to closely adopt and sometimes directly memorize the style and the content of different training images – defined as "concepts" in the training data [11, 21]. This leads to concerns from creatives whose work has been used to train GenAI. Concerns focus upon the lack of a means for attribution, *e.g.* recognition or citation, of synthetic images to the training data used to create them and extend even to calls for a compensation mechanism (financial, reputational, or otherwise) for GenAI's derivative use of concepts in training images contributed by creatives.

We refer to this problem as *concept attribution* – the ability to attribute generated images to the training concept/s which have most directly influenced their creation. Several

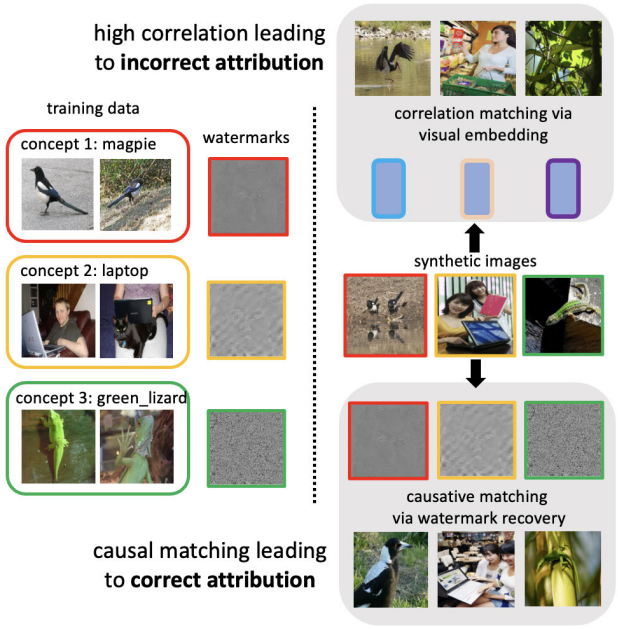

Figure 1. **Causative *vs*. correlation-based matching for concept attribution.** ProMark identifies the training data most responsible for a synthetic image ('attribution'). Correlation-based matching doesn't always perform the data attribution properly. We propose ProMark, which is a proactive approach involving adding watermarks to training data and recovering them from the synthetic image to perform attribution in a causative way.

passive techniques have recently been proposed to solve the attribution problem [5, 30, 34]. These approaches use visual correlation between the generated image and the training images for attribution. Whilst they vary in their method and rationale for learning the similarity embedding – all use some forms of contrastive training to learn a metric space for visual correlation.

We argue that although correlation can provide visually intuitive results, a measure of similarity is not a causative answer to whether certain training data is responsible for the generation of an image or not. Further, correlation-based techniques can identify close matches with images that were not even present in the training data.

Keeping this in mind, we explore an intriguing field

of research which is developing around proactive watermarking methodologies [3, 29, 33, 37], that employ signals, termed *templates* to encrypt input images before feeding them into the network. These works have integrated and subsequently retrieved templates to bolster the performance of the problem at hand. Inspired by these works, we introduce ProMark, a proactive watermarking-based approach for GenAI models to perform concept attribution in a causative way. The technical contributions of ProMark are three-fold:

1. Causal vs. Correlation-based Attribution. ProMark performs causal attribution of synthetic images to the predefined concepts in the training images that influenced the generation. Unlike prior works that visually correlate synthetic images with training data, we make no assumption that visual similarity approximates causation. ProMark ties watermarks to training images and scans for the watermarks in the generated images, enabling us to demonstrate rather than approximate/imply causation. This provides confidence in grounding downstream decisions such as legal attribution or payments to creators.

2. Multiple Orthogonal Attributions. We propose to use orthogonal invisible watermarks to proactively embed attribution information into the input training data and add a BCE loss during the training of diffusion models to retain the corresponding watermarks in the generated images. We show that ProMark causatively attributes as many as 2^{16} unique training-data concepts like objects, scenes, templates, motifs, and style, where the generated images can simultaneously express one or two orthogonal concepts.

3. Flexible Attributions. ProMark can be used for training conditional or unconditional diffusion models and even finetuning a pre-trained model for only a few iterations. We show that ProMark's causative approach achieves higher accuracy than correlation-based attribution over five diverse datasets (Sec. 4.1): Adobe Stock, ImageNet, LSUN, Wikiart, and BAM while preserving synthetic image quality due to the imperceptibility of the watermarks.

Fig. 1 presents our scenario, where synthetic image(s) are attributed back to the most influential GenAI training images. Correlation-based techniques [5, 34] try to match the high-level image structure or style. Here, the green-lizard synthetic image is matched to a generic green image without a lizard [5]. With ProMark's causative approach, the presence of the green-lizard watermark in the synthetic image will correctly indicate the influence of the similarly watermarked concept group of lizard training images.

2. Related Works

Passive Concept Attribution. Concept attribution differs from model [8] or camera [12] attribution in that the task is to determine the responsible training data for a given generation. Existing concept attribution techniques are passive – they do not actively modify the GenAI model or training data but instead, measure the visual similarity (under some definition) of synthetic images and training data to quantify attribution for each training image. EKILA [5] proposes patch-based perceptual hashing (visual fingerprinting [6, 24]) to match query patches to the training data for attribution. Wang *et al*. [34] propose Attribution by Customization (AbC), calibrating embeddings like CLIP, DINO, *etc*. for the attribution task using images generated from "customized" diffusion models in the training loop. Both [5] and [34] also explored ALADIN [30] for style attribution; a feature for fine-grained style similarity. All these approaches approximate causation by visual correlation within a contrastively trained embedding. They are passive approaches that take the image as an attribute by correlating between generated and training images. Instead, our approach is a proactive scheme that adds a watermark to training images and performs attribution in a causal manner (Tab. 1).

Table 1. **Comparison of ProMark with prior works**. Uniquely, we perform causative attribution using proactive watermarking to attribute multiple concepts. [Keys: emb.: embedding, obj.: object, own.: ownership, sem.: semantic, sty.: style, wat.: watermark]

Method	Scheme type	Task	Match type	# Class	Multiple attribution	Attribution type
[30]	passive	attribution	emb.	-	×	sty.
[5]	passive	attribution	emb.	-	×	obj.
[34]	passive	attribution	emb.	693	×	sty., obj.
[17]	passive	detect	wat.	2	×	-
[22]	passive	detect	wat.	2	×	-
[14]	passive	detect	wat.	2	×	-
[33]	proactive	detect	wat.	2	-	-
[1]	proactive	detect	wat.	2	-	-
[3]	proactive	localization	wat.	2	-	-
[2]	proactive	obj. detect	-	90	-	-
ProMark	proactive	attribution	wat.	2^{16}	✓	sty., obj. own., sem.

Leave-One-Out Training. Early works retrained models holding out training data to determine its influence [16, 20]. Whilst causal, the lengthy training required for GenAI models is not practical for our image attribution task.

Proactive Schemes. Proactive schemes involve adding a signal/perturbation onto the input images to benefit different tasks like deepfake tagging [33], deepfake detection [1], manipulation localization [3], object detection [2], *etc*. Some works [29, 37] disrupt the output of the generative models by adding perturbations to the training data. Alexandre *et al*. [31] tackles the problem of training dataset attribution by using fixed signals for every data type. These prior works successfully demonstrate the use of watermarks to classify the content of the AI-generated images proactively. We extend the idea of proactive watermarking to perform the task of causal attribution of AI-generated images to influential training data concepts. Watermarking has not been used to trace attribution in GenAI before.

Watermarking of GenAI Models. It is an active research to watermark AI-generated images for the purpose of privacy protection. Fernandez *et al*. [17] fine-tune the LDM's

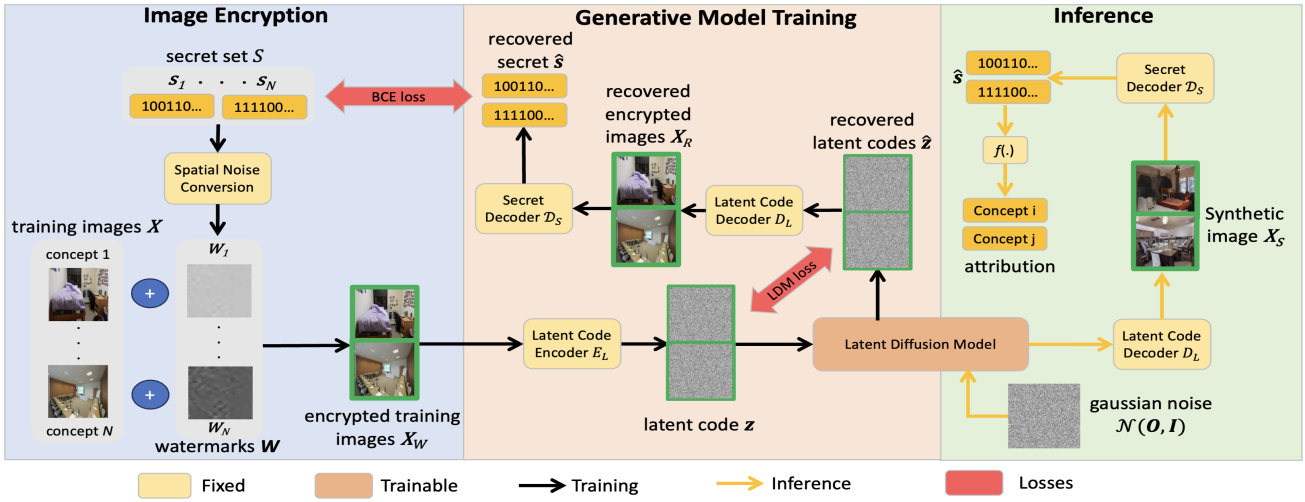

Figure 2. **Overview of ProMark**. We show the training and inference procedure for our proposed method. Our training pipeline involves two stages, image encryption and generative model training. We convert the bit-sequences to spatial watermarks ($\boldsymbol{W}$), which are then added to the corresponding concept images ($\boldsymbol{X}$) to make them encrypted ($\boldsymbol{X}_W$). The generative model is then trained with the encrypted images using the LDM supervision. During training, we recover the added watermark using the secret decoder ($\mathcal{D}_S$) and apply the BCE supervision to perform attribution. To sample newly generated images, we use a Gaussian noise and recover the bit-sequences using the secret decoder to attribute them to different concepts. Best viewed in color.

decoder to condition on a bit sequence, embedding it in images for AI-generated image detection. Kirchenbauer *et al*. [19] propose a watermarking method for language models by pre-selecting random tokens and subtly influencing their use during word generation. Zhao *et al*. [39] use a watermarking scheme for text-to-image diffusion models, while Liu *et al*. [22] verify watermarks by pre-defined prompts. [14, 25] add a watermark for detecting copyright infringement. Asnani *et al*. [4] reverse engineer a fingerprint left by various GenAI models to further use it for recovering the network and training parameters of these models [4, 36]. Finally, Cao *et al*. [10] adds an invisible watermark for protecting diffusion models which are used to generate audio modality. Most of these works have used watermarking for protecting diffusion models, which enables them to add just one watermark onto the data. In contrast, we propose to add multiple watermarks to the training data and to a single image, which is a more challenging task than embedding a universal watermark.

3. Method

3.1. Background

Diffusion Models. Diffusion models learn a data distribution $p(\boldsymbol{X})$, where $\boldsymbol{X} \in \mathbb{R}^{h \times w \times 3}$ is the input image. They do this by iteratively reducing the noise in a variable that initially follows a normal distribution. This can be viewed as learning the reverse steps of a fixed Markov Chain with a length of T. Recently, LDM [28] is proposed to convert images to their latent representation for faster training in a lower dimensional space than the pixel space. The image is converted to and from the latent space by a pretrained autoencoder consisting of an encoder $\boldsymbol{z} = \mathcal{E}_L(\boldsymbol{X})$ and a decoder $\boldsymbol{X}_R = \mathcal{D}_L(\boldsymbol{z})$, where $\boldsymbol{z}$ is the latent code and $\boldsymbol{X}_R$ is the reconstructed image. The trainable denoising module of the LDM is $\epsilon_\theta(\boldsymbol{z}_t, t); t = 1...T$, where ϵ_θ is trained to predict the denoised latent code $\hat{\boldsymbol{z}}$ from its noised version $\boldsymbol{z}_t$. This objective function can be defined as follows:

$$L_{LDM} = \mathbb{E}_{\mathcal{E}_L(\boldsymbol{X}), \epsilon \sim \mathcal{N}(0,1), t}\left[||\epsilon - \epsilon_\theta(\boldsymbol{z}_t, t)||_2^2\right], \quad (1)$$

where ϵ is the noise added at step t.

Image Encryption. Proactive works [1–3] have shown performance gain on various tasks by proactively transforming the input training images $\boldsymbol{X}$ with a watermark, resulting in an encrypted image. This watermark is either fixed or learned, depending on the task at hand. Similar to prior proactive works, our image encryption is of the form:

$$\boldsymbol{X}_W = \mathcal{T}(\boldsymbol{X}; \boldsymbol{W}) = \boldsymbol{X} + m \times R(\boldsymbol{W}, h, w), \quad (2)$$

where $\mathcal{T}$ is the transformation, $\boldsymbol{W}$ is the spatial watermark, $\boldsymbol{X}_W$ is the encrypted image, m is the watermark strength, and $R(.)$ resizes $\boldsymbol{W}$ to the input resolution (h, w).

We use the state-of-the-art watermarking technique RoSteALS [9] to compute the spatial watermarks for encryption due to its robustness to image transformation and generalization (the watermark is independent of content of the input image). RoSteALS is designed to embed a secret of length b-bits into an image using robust and imperceptible watermarking. It comprises of a secret encoder $\mathcal{E}_S(\boldsymbol{s})$, which converts the bit-secret $\boldsymbol{s} \in \{0, 1\}^b$ into a latent code offset $\boldsymbol{z}_o$. It is then added to the latent code of an autoencoder $\boldsymbol{z}_w = \boldsymbol{z} + \boldsymbol{z}_o$. This modified latent code $\boldsymbol{z}_w$ is then used to

reconstruct a watermarked image via autoencoder decoder. Finally, a secret decoder, denoted by $\mathcal{D}_S(X_W)$, takes the watermarked images as input and predicts bit-sequence $\hat{\boldsymbol{s}}$.

3.2. Problem Definition

Let $\mathcal{C} = \{c_1, c_2, \ldots, c_N\}$ be a set of N distinct concepts within a dataset that is used for training a GenAI model for image synthesis. The problem of concept attribution can be formulated as follows:

Given a synthetic image $\boldsymbol{X}_S$ generated by a GenAI model, the objective of concept attribution is to accurately associate $\boldsymbol{X}_S$ to a concept $c_i \in \mathcal{C}$ that significantly influenced the generation of $\boldsymbol{X}_S$.

We aim to find a mapping $f : \boldsymbol{X}_S \rightarrow c_i$ such that

$$c_i^* = \arg\max_{c_i \in \mathcal{C}} f(\boldsymbol{X}_S, c_i), \tag{3}$$

where c_i^* represents the concept most strongly attributed to image $\boldsymbol{X}_S$.

3.3. Overview

The pipeline of ProMark is shown in Fig. 2. The principle is simple: if a specific watermark unique to a training concept can be detected from a generated image, it indicates that the generative model relies on that concept in the generation process. Thus, ProMark involves two steps: training data encryption via watermarks and generative model training with watermarked images.

To watermark the training data, the dataset is first divided into N groups, where each group corresponds to a unique concept that needs attribution. These concepts can be semantic (*e.g.* objects, scenes, motifs or stock image templates) or abstract (*e.g.* stylistic, ownership info). Each training image in a group is encoded with a unique watermark without significantly altering the image's perceptibility. Once the training images are watermarked, they are used to train the generative model. As the model trains, it learns to generate images based on the encrypted training images. Ideally, the generated images would have traces of watermarks corresponding to concepts they're derived from.

During inference, ProMark conforms to whether a generated image is derived from a particular training concept by identifying the unique watermark of that concept within the image. Through the careful use of unique watermarks, we can trace back and causally attribute generated images to their origin in the training dataset.

3.4. Training

During training, our algorithm is composed of two stages: image encryption and generative model training. We now describe each of these stages in detail.

Image Encryption. The training data is first divided into N concepts, and images in each partition are encrypted using a fixed spatial watermark $\boldsymbol{W}_j \in \mathbb{R}^{h \times w}$ ($j \in 0, 1, 2, ..., N$). Each noise $\boldsymbol{W}_j$ is a b-dim bit-sequence (secret) $\boldsymbol{s}_j = \{p_{j1}, p_{j2}, ..., p_{jb}\}$ where $p_{ji} \in \{0, 1\}$.

In order to compute the watermark $\boldsymbol{W}_j$ from the bit-sequence $\boldsymbol{s}_j$, we encrypt 100 random images with $\boldsymbol{s}_j$ using pretrained RoSteALS secret encoder $\mathcal{E}_S(.)$ which takes $b = 160$ length secret as input. From these encrypted images, we obtain 100 noise residuals by subtracting the encrypted images from the originals, which are averaged to compute the watermark $\boldsymbol{W}_j$ as:

$$\boldsymbol{W}_j = \frac{1}{100} \sum_{i=1}^{100} (\boldsymbol{X}_i - \mathcal{E}_S(\boldsymbol{X}_i, \boldsymbol{s}_j)). \tag{4}$$

The above process is defined as spatial noise conversion in Fig. 2. The averaging of noise residuals across different images reduces the image content in the watermark and makes the watermark independent of any specific image. Additionally, the generated watermarks are orthogonal due to different bits for all $\boldsymbol{s}_j$, ensuring distinguishability from each other. With the generated watermarks, each training image is encrypted using Eq. (2) with one of the N watermarks that correspond to the concept the image belongs to.

Generative Model Training. Using the encrypted data, we train the LDM's denoising module $\epsilon_\theta(.)$ using the objective function (Eq. (1)), where $\boldsymbol{z}_t$ is the noised version of:

$$\boldsymbol{z} = \mathcal{E}_L(\boldsymbol{X}_{W_j}) = \mathcal{E}_L(\mathcal{T}(\boldsymbol{X}; \boldsymbol{W}_j)), \tag{5}$$

i.e., the input latent codes $\boldsymbol{z}$ are generated using the encrypted images $\boldsymbol{X}_{W_j}$ for $j \in \{0, 1, 2...., N\}$.

However, we found that only using LDM loss is insufficient to successfully learn the connection between the conceptual content and its associated watermark. This gap in learning presents a significant hurdle, as the primary aim is to trace back generated images to their respective training concepts via the watermark. To tackle this, an auxiliary supervision is introduced to LDM's training,

$$L_{BCE}(\boldsymbol{s}_j, \hat{\boldsymbol{s}}) = -\frac{1}{b} \sum_{i=1}^{b} [p_{ji} \log(\hat{p}_i) + (1 - p_{ji}) \log(1 - \hat{p}_i)], \tag{6}$$

where $L_{BCE}(.)$ is the binary cross-entropy (BCE) between the actual bit-sequence $\boldsymbol{s}_j$ associated with watermark $\boldsymbol{W}_j$ and the predicted bit-sequence $\hat{\boldsymbol{s}}$. The denoised latent code $\hat{\boldsymbol{z}}$ is then decoded using the autoencoder $\mathcal{D}_L(.)$, and the embedded secret $\hat{\boldsymbol{s}}$ is predicted by the secret decoder $\mathcal{D}_S(.)$ as:

$$\hat{\boldsymbol{s}} = \mathcal{D}_S(\mathcal{D}_L(\hat{\boldsymbol{z}})). \tag{7}$$

By employing BCE, the model is guided to minimize the difference between the predicted watermark and the embedded watermark, hence improving the model's ability to associate watermarks with their respective concepts. Finally, our objective is to minimize the loss $L_{attr} = L_{LDM} + \alpha L_{BCE}$ during training, where $\alpha = 2$ for our experiments.

3.5. Inference

After the LDM learns to associate the watermarks with concepts, we use random Gaussian noise to sample the newly generated images from the model. While the diffusion model creates these new images, it also embeds a watermark within them. Each watermark maps to a distinctive orthogonal bit-sequence associated with a specific training concept, serving as a covert signature for attribution.

To attribute the generated images and ascertain which training concept influenced them, we predict the secret embedded by the LDM in the generated images (see Eq. (7)). Given a predicted binary bit-sequence, $\hat{\boldsymbol{s}} = \{\hat{p_1}, \hat{p_2}, ..., \hat{p_b}\}$ and all the input bit-sequences $\boldsymbol{s}_j$ for $j \in 0, 1, 2..., N$, we define the attribution function, f, in Eq. (3) as:

$$f(\hat{\boldsymbol{s}}, \boldsymbol{s}_j) = \sum_{k=1}^{b} [\hat{p}_k = p_{jk}], \tag{8}$$

where $[\hat{p}_k = p_{ik}]$ acts as an indicator function, returning 1 if the condition is true, *i.e.*, the bits are identical, and 0 otherwise. Consequently, we assign the predicted bit sequence to the concept whose bit sequence it most closely mirrors — that is, the concept j^* for which $f(\hat{\boldsymbol{s}}, \boldsymbol{s}_{j^*})$ is maximized:

$$j^* = \arg \max_{j \in \{1,2,...,N\}} f(\hat{\boldsymbol{s}}, \boldsymbol{s}_j). \tag{9}$$

In other words, the concept whose watermark is most closely aligned with the generated image's watermark is deemed to be the influencing source behind the generated image.

3.6. Multiple Watermarks

In prior image attribution works, an image is usually attributed to a single concept (*e.g.* image content or image style). However, in real-world scenarios, an image may encapsulate multiple concepts. This observation brings forth a pertinent question: "Is it possible to use multiple watermarks for multi-concept attribution within a single image?"

In this paper, we propose a novel approach to perform multi-concept attribution by embedding multiple watermarks into a single image. In our preliminary experiments, we restrict our focus to the addition of two watermarks. To achieve this, we divide the image into two halves and resize each watermark to fit the respective halves. This ensures that each half of the image carries distinct watermark information pertaining to a specific concept.

For the input RGB image $\boldsymbol{X}$, $\{\boldsymbol{W}_i, \boldsymbol{W}_j\}$ are the watermarks for two secrets $\{\boldsymbol{s}_i, \boldsymbol{s}_j\}$, we formulate the new transformation $\mathcal{T}$ as:

$$\begin{aligned}\mathcal{T}(\boldsymbol{X}; \boldsymbol{W}_i, \boldsymbol{W}_j) &= \Big\{ \boldsymbol{X}_{left}, \boldsymbol{X}_{right} \Big\} \\ &= \Big\{ (\boldsymbol{X}(:, 0 : \frac{w}{2}, :) + R(\boldsymbol{W}_i, h, \frac{w}{2})), \\ &\quad (\boldsymbol{X}(:, \frac{w}{2} : w, :) + R(\boldsymbol{W}_j, h, \frac{w}{2}) \Big\},\end{aligned}$$

Table 2. Comparison with prior works for unconditional diffusion model on various datasets. [Keys: str.: strength]

Method	Str. (%)	Attribution Accuracy (%) ↑				
		Stock	LSUN	Wiki-A	Wiki-S	ImageNet
ALADIN [30]	-	99.86	46.27	48.95	33.25	9.25
CLIP [27]	-	75.67	87.13	77.58	60.84	60.12
AbC-CLIP [34]	-	78.49	87.39	77.23	60.43	62.83
SSCD [26]	-	99.63	73.26	69.51	50.37	37.32
EKILA [5]	-	99.37	70.60	51.23	37.06	38.00
ProMark	30	100	95.12	97.45	98.12	83.06
	100	**100**	**100**	**100**	**100**	**91.07**

where $\{.\}$ is the horizontal concatenation. The loss function uses the two predicted secrets ($\hat{\boldsymbol{s}}_1$ and $\hat{\boldsymbol{s}}_2$) from the two halves of the generated image, defined as:

$$L_{attr} = L_{LDM} + \alpha(L_{BCE}(\boldsymbol{s}_i, \hat{\boldsymbol{s}}_1) + L_{BCE}(\boldsymbol{s}_j, \hat{\boldsymbol{s}}_2)).$$

4. Experiments

4.1. Unconditional Diffusion Model

In this section, we train multiple versions of unconditional diffusion models [28] to demonstrate that ProMark can be used to attribute a variety of concepts in the training data. In each case, the model is trained starting from random initialization of LDM weights. Described next are details of the datasets and evaluation protocols.

Datasets We use 5 datasets spanning attribution categories like image templates, scenes, objects, styles, and artists. For each dataset, we consider the dataset classes as our attribution categories. For each class in a dataset, we use 90% images for training, and 10% for evaluation, unless specified otherwise.

1. Stock: We collect images from Adobe Stock, comprising of near-duplicate image clusters like templates, symbols, icons, *etc*. An example image from some clusters is shown in the supplementary. We use 100 such clusters, each with $2K$ images.
2. LSUN: The LSUN dataset [38] comprises 10 scene categories, such as bedrooms and kitchens. It's commonly used for scene classification, training generative models like GANs, and anomaly detection. Same as the Stock dataset, we use $2K$ images per class.
3. Wiki-S: The WikiArt dataset [32] is a collection of fine art images spanning various styles and artists. We use the 28 style classes with 580 average images per class.
4. Wiki-A: From the WikiArt dataset [32] we also use the 23 artist classes with 2,112 average images per class.
5. ImageNet: We use the ImageNet dataset [15] which comprises of 1 million images across $1K$ classes. For this dataset, we use the standard validation set with $50K$ for evaluation and the remaining images for training.

Evaluation Protocol For all datasets, the concept attribution performance is tested on the held-out data as follows. For a held-out image, we first encrypt it with the concept's watermark. Then using the latent code of the encrypted image, we noise it till a randomly assigned timestamp and ap-

Figure 3. **Example training and newly sampled images of different datasets for the corresponding classes**. We observe a similar content in the inference image compared with the training image of the predicted class.

Figure 4. **Visual results of prior embedding-based works.** We show the image of the closest matched embedding for each method on ImageNet. We highlight images green for correct attribution, otherwise red. Embedding-based works do not always attribute to the correct concept.

ply our trained diffusion model to reverse back to the initial timestamp with the estimated noise. The denoised latent code is then decoded using the autoencoder $\mathcal{D}_L(.)$, and the embedded secret is predicted using the secret decoder $\mathcal{D}_S(.)$. Using Eq. (9), we compute the predicted concept and calculate the accuracy using the ground-truth concept.

Results Shown in Tab. 2 is the attribution accuracy of ProMark at two watermark strengths *i.e.* 100% and 30% which is set by variable m in Eq. (2). ProMark outperforms prior works, achieving near-perfect accuracy on all the datasets when the watermark strength is 100%. However, the watermark introduces visual artifacts [9] if the watermark strength is full. Therefore, we decrease the watermark strength to 30% before adding it to the training data (see Sec. 4.5 for ablation on watermark strength). Even though our performance drops at a lower watermark strength, we still outperform the prior works. This shows that our causal approach can be used to attribute a variety of concepts in the training data with an accuracy higher than the prior passive approaches.

Fig. 3 (rows 1-5) shows the qualitative examples of the newly sampled images from each of the trained models. For each model, we sample the images using random Gaussian noise until we have images for every concept. The concept for each image is predicted using the secret embedded in the generated images. Shown in each row of Fig. 3 are three training images (columns 1-3) and three sampled images from the corresponding concepts (columns 4-6). This shows that ProMark makes the diffusion model embed the corresponding watermark for the class of the generated image, thereby demonstrating the usefulness of our approach.

Shown in Fig. 4 are the nearest images retrieved using the embedding-based methods (row (2)-(6)) for the query images from the ImageNet(row (1)). For each image retrieval, we highlight the correct/incorrect attribution using a green/red box. As we can see, the correlation-based prior techniques rely on visual similarity between the query and the retrieved images, ignoring the concept. However, for each query image, ProMark predicts the correct concept corresponding to the query image (Fig. 3).

4.2. Multiple Watermarks

We evaluate the effectiveness of ProMark for multi-concept attribution. As before, an unconditional diffusion model is trained starting from random initialization, and each image in the training data is encrypted with two watermarks as outlined in Sec. 3.6.

Dataset For this experiment, we use the BAM dataset [35], comprising contemporary artwork sourced from Behance, a platform hosting millions of portfolios by professionals and artists. This dataset uniquely categorizes each image into two label types: media and content. It encompasses 7 distinct labels for media and 9 for content, culminating in a diverse set of 63 label pairs, with $4,593$ average images in these label pairs. For each class pair, we use 90% data for

Table 3. Multi-concept attribution comparison with baselines.

Method	Strength (%)	Attribution Accuracy (%) ↑		
		Media	Content	Combined
ALADIN [30]	-	42.16	41.25	34.97
CLIP [27]	-	46.71	45.12	42.36
AbC-CLIP [34]	-	52.12	51.56	46.23
SSCD [26]	-	47.06	46.09	40.61
EKILA [5]	-	43.72	43.58	37.09
ProMark (single)	30	-	-	**97.73**
ProMark (multi)	30	91.33	89.21	84.66
	100	**95.61**	**93.31**	90.12

training and 10% for held-out evaluation.

Results The same evaluation is performed as described in Sec. 4.1, except the accuracy is now computed for two concepts instead of one. Shown in Tab. 3 is the attribution accuracy for the two concepts individually and simultaneously. To benchmark the effectiveness of ProMark, we also compare against baselines, where ProMark outperforms baselines, achieving a combined attribution accuracy of 90.12% as compared to 46.61% for AbC-CLIP [34] (Attribution by Customization, fine-tuning of CLIP). We believe our findings substantiate that ProMark can be extended to a scenario where the generated images are composed of several unique concepts from the training images. For ablation, we train ProMark with 7×8 classes, with each pair of media and content as an individual concept. ProMark is able to achieve 97.73% attribution accuracy for single-concept, higher than the performance achieved for multi-concept case *i.e.* 90.12%. However, single concept approach is not scalable when the number of concepts in an image increases, as the number of watermarks would grow exponentially (7×8 *vs.* $7 + 8$). Therefore, transitioning to a multi-concept scenario is more appropriate for real-world scenarios, where scalability and practicality are crucial.

In the final row of Fig. 3, we present qualitative examples of newly sampled images from the model trained on the BAM dataset. Observations indicate that these sampled images successfully adopt both media and content corresponding to training images of the same concept. This provides empirical evidence of ProMark's effectiveness in facilitating multi-concept attribution.

4.3. Number of Concepts

AI models leverage large-scale image datasets [7, 18, 23, 28], encompassing a broad spectrum of concepts. This diversity necessitates concept attribution methods that can maintain high performance across numerous concepts. In this context, we test ProMark with an exponentially increasing number of concepts. Our dataset comprises Adobe Stock images with near duplicate image templates (used as concepts). As we escalate the number of concepts, we concurrently reduce the per-concept image count, only 24 images per concept for 2^{16} concepts, see the red curve of Fig. 5 (a) for image count. This is done to obtain balanced image distribution and also to challenge ProMark's robustness.

Table 4. Comparison with different baselines for the conditional model trained on ImageNet dataset.

Method	Strength (%)	Attribution Accuracy (%) ↑	
		Held-out data	New images
ALADIN [30]	-	9.25	0.18
CLIP [27]	-	60.12	41.01
AbC-CLIP [34]	-	62.83	50.19
SSCD [26]	-	37.32	30.10
EKILA [5]	-	38.00	29.06
ProMark	30	91.24	87.30
	100	**95.60**	**90.13**

The outcomes, depicted in Fig. 5(a) red curve, indicate an anticipated decline in ProMark's efficacy in line with the increase in the number of concepts, reducing from 100% attribution accuracy for 10 concepts (chance accuracy 10%) to 82% for 2^{16} concepts (chance accuracy $1.5e$-3%). This reduction in attribution accuracy is correlated with the reduction in bit-secret accuracy (green curve) for every predicted secret, indicating poor watermark recovery due to the increased confusion between the watermarks. Notwithstanding the increased difficulty, ProMark demonstrates commendable performance, underscoring its potential in real-world applications.

4.4. Conditional Diffusion Model

As the diffusion models are usually trained with conditions to guide generation, we also evaluate using the conditional LDM model [28]. For this, we fine-tune a model pretrained of the ImageNet dataset (see Sec. 4.1), where the 1000 ImageNet classes are used as model conditions and also as the 1000 concepts.

Evaluation Protocol In addition to the evaluation on the held-out data (see Sec. 4.1), we also perform the quantitative evaluation on the newly sampled images as follows. We use the labels of the ImageNet dataset as conditions to sample $10K$ images (10 images per label). Using these labels as the ground-truth concept for a newly sampled image, we compute the accuracy of the concept predicted by the embedded watermark in the generated images.

Results The accuracies for held-out and newly sampled images are shown in Tab. 4. The performance on the held-out dataset for the conditional model improves compared to the unconditional models as the label conditions provide improved supervision for correct watermarks. ProMark also outperforms prior embedding-based works by a large margin on both held-out and newly sampled images. The attribution accuracy on the new images, however, is less than the held-out data. We hypothesize that it is because newly sampled images may contain more than one concept and can be more confusing to attribute. The high accuracy, even for newly sampled images, suggests that ProMark exhibits higher generalizability to unseen synthetic images.

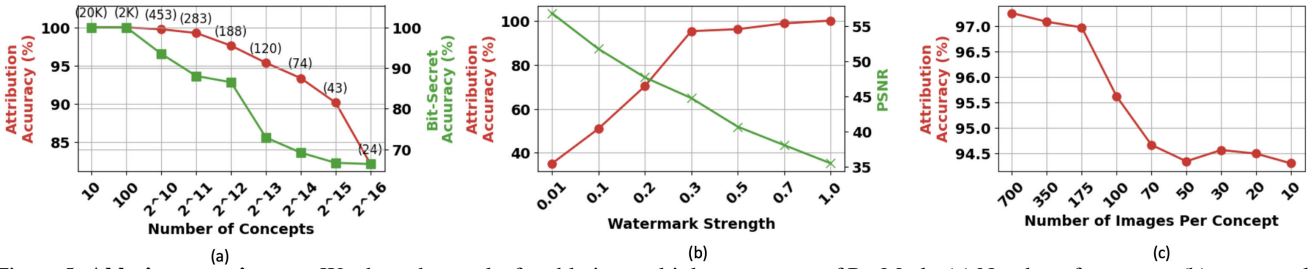

Figure 5. **Ablation experiments**: We show the results for ablating multiple parameters of ProMark. (a) Number of concepts, (b) watermark strength, and (c) number of images per concept.

4.5. Ablation Study

For the ablation experiment, we use Stock dataset with a varying number of concepts, and we train unconditional LDM models from random initialization.

Strength of Watermark. The hyperparameter m in Eq. (2) modulates the intensity of the watermark applied to the training images, ensuring encrypted images retain high quality. We systematically alter m to examine its impact on the LDM's performance and the Peak Signal-to-Noise Ratio (PSNR) of the output images with reference to the held-out encrypted images. Fig. 5(b) shows that attribution accuracy improves with increased m, plateauing beyond a threshold of 0.5. The discernible compromise in image quality, as evidenced by the inverse relationship between intensity and PSNR, can be attributed to the use of fixed watermarks obtained using RoSteALS [9], which is originally optimized for robustness. In light of this, we select an optimal watermark strength of 0.3, which balances between performance and PSNR. We measured the FID between original and newly sampled images from a pretrained ImageNet conditional model (trained without watermark) and ProMark model (trained with watermark), which is 13.28 and 17.63 respectively. This small increment shows negligible quality loss in the generated images due to ProMark.

Number of Images Per Concept. To ascertain the optimal number of images required per concept for effective watermark learning, we ablate by fixing the number of concepts to 500 and varying the number of images used to train the LDM. Fig. 5(c) reveals that performance drops by 2.5% when image count per concept is reduced from 700 to 10. Remarkably, the general efficacy of ProMark remains consistently high, suggesting a low sensitivity to the image count per concept. These results demonstrate that ProMark can successfully learn watermarks with as few as 10 images per concept, highlighting its efficiency and potential for applications with limited data availability.

Framework Design. ProMark employs BCE loss to instruct the LDM model in the accurate embedding of bit-sequence watermarks within generated images. The attribution performance degrades to 2% when BCE loss is not used as compared to 100% in Tab. 2. This shows that removing BCE loss significantly impairs the LDM's performance, underscoring the necessity of this supervision in helping LDM embed watermarks effectively.

Also, ProMark incorporates a secret decoder to retrieve secret bit-sequence from synthesized images, rendering the process contingent upon the pretrained secret decoder. In contrast, prior works [1–3] recover watermarks by training a dedicated decoder with the main model in an end-to-end fashion. To ablate this alternative approach, we train a standard decoder along with LDM by optimizing for the cosine similarity between the embedded and extracted watermarks. We see a degradation in performance from 100% to 80.56%, indicating that the pretrained secret decoder is a better choice for our approach. This is due to the increased complexity of predicting watermarks of resolution 256^2 as compared to 160-bit sequence from the encrypted images.

5. Conclusion

We introduce a novel proactive watermarking-based approach, ProMark, for causal attribution. We use predefined training concepts like styles, scenes, objects, motifs, *etc*. to attribute the influence of training data on generated images. We show ProMark's is effective across various datasets and model types, maintaining image quality while providing more accurate attribution on a large number of concepts. Our approach can also be extended to multi-concept attribution by embedding multiple watermarks onto the image. Finally, for each experiment, our approach achieves a higher attribution accuracy than the prior passive approaches. Such attribution offers opportunities to recognize and reward creative contributions to generative AI, underpinning new models for value creation in the future creative economy [13].

Limitations. In evaluating ProMark, we note a trade-off between image quality and attribution accuracy, which may need us to learn watermarks for attribution task. Our model is currently trained with predefined concepts and further research is needed on training paradigm when new concepts are introduced. While we use orthogonal watermarks for varied concepts like motifs and styles, this may not accurately reflect the interrelated nature of some concepts, suggesting another opportunity for future research. Finally, our results are specific to the LDM, and extending this approach to other GenAI models could provide a better understanding of ProMark's effectiveness.

References

[1] Vishal Asnani, Xi Yin, Tal Hassner, Sijia Liu, and Xiaoming Liu. Proactive image manipulation detection. In *CVPR*, 2022. 2, 3, 8

[2] Vishal Asnani, Abhinav Kumar, Suya You, and Xiaoming Liu. PrObeD: Proactive object detection wrapper. In *NeurIPS*, 2023. 2

[3] Vishal Asnani, Xi Yin, Tal Hassner, and Xiaoming Liu. MaLP: Manipulation localization using a proactive scheme. In *CVPR*, 2023. 2, 3, 8

[4] Vishal Asnani, Xi Yin, Tal Hassner, and Xiaoming Liu. Reverse engineering of generative models: Inferring model hyperparameters from generated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15477–15493, 2023. 3

[5] Kar Balan, Shruti Agarwal, Simon Jenni, Andy Parsons, Andrew Gilbert, and John Collomosse. EKILA: Synthetic media provenance and attribution for generative art. In *CVPR*, 2023. 1, 2, 5, 7

[6] Alex Black, Tu Bui, Hailin Jin, Vishy Swaminathan, and John Collomosse. Deep image comparator: Learning to visualize editorial change. In *CVPR WMF*, 2021. 2

[7] Andreas Blattmann, Robin Rombach, Kaan Oktay, Jonas Müller, and Björn Ommer. Retrieval-augmented diffusion models. *NeurIPS*, 2022. 7

[8] Tu Bui, Ning Yu, and John Collomosse. RepMix: Representation mixing for robust attribution of synthesized images. In *ECCV*, 2022. 2

[9] Tu Bui, Shruti Agarwal, Ning Yu, and John Collomosse. RoSteALS: Robust steganography using autoencoder latent space. In *CVPR*, 2023. 3, 6, 8

[10] Xirong Cao, Xiang Li, Divyesh Jadav, Yanzhao Wu, Zhehui Chen, Chen Zeng, and Wenqi Wei. Invisible watermarking for audio generation diffusion models. In *TPS-ISA*, 2023. 3

[11] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX*, 2019. 1

[12] Chang Chen, Zhiwei Xiong, Xiaoming Liu, and Feng Wu. Camera trace erasing. In *CVPR*, 2020. 2

[13] John Collomosse and Andy Parsons. To Authenticity, and Beyond! Building safe and fair generative AI upon the three pillars of provenance. *IEEE Computer Graphics and Applications*, 2024. 8

[14] Yingqian Cui, Jie Ren, Han Xu, Pengfei He, Hui Liu, Lichao Sun, and Jiliang Tang. DiffusionShield: A watermark for copyright protection against generative diffusion models. *arXiv preprint arXiv:2306.04642*, 2023. 2, 3

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

[16] Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In *Proc. NeurIPS*, 2020. 2

[17] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *ICCV*, 2023. 2

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 1, 7

[19] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *ICML*, 2023. 3

[20] Pang Koh and Percy Liang. Understanding black-box predictions via influence functions. In *Proc. ICML*, 2017. 2

[21] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *ICCV*, 2023. 1

[22] Yugeng Liu, Zheng Li, Michael Backes, Yun Shen, and Yang Zhang. Watermarking diffusion model. *arXiv preprint arXiv:2305.12502*, 2023. 2, 3

[23] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *CVPR*, 2023. 1, 7

[24] Eric Nguyen, Tu Bui, Vishy Swaminathan, and John Collomosse. OSCAR-Net: Object-centric scene graph attention for image attribution. In *ICCV*, 2021. 2

[25] Sen Peng, Yufei Chen, Cong Wang, and Xiaohua Jia. Protecting the intellectual property of diffusion models by the watermark diffusion process. *arXiv preprint arXiv:2306.03436*, 2023. 3

[26] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *CVPR*, 2022. 5, 7

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 5, 7

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 3, 5, 7

[29] Nataniel Ruiz, Sarah Adel Bargal, and Stan Sclaroff. Disrupting deepfakes: Adversarial attacks against conditional image translation networks and facial manipulation systems. In *ECCVW*, 2020. 2

[30] Dan Ruta, Saeid Motiian, Baldo Faieta, Zhe Lin, Hailin Jin, Alex Filipkowski, Andrew Gilbert, and John Collomosse. ALADIN: All layer adaptive instance normalization for fine-grained style similarity. In *ICCV*, 2021. 1, 2, 5, 7

[31] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Radioactive data: tracing through training. In *ICML*, 2020. 2

[32] Wei Ren Tan, Chee Seng Chan, Hernan Aguirre, and Kiyoshi Tanaka. Improved ArtGAN for conditional synthesis of natural image and artwork. *IEEE Transactions on Image Processing*, 28(1):394–409, 2019. 5

[33] Run Wang, Felix Juefei-Xu, Meng Luo, Yang Liu, and Lina Wang. FakeTagger: Robust safeguards against deepfake dissemination via provenance tracking. In *ACM MM*, 2021. 2

[34] Sheng-Yu Wang, Alexei A Efros, Jun-Yan Zhu, and Richard Zhang. Evaluating data attribution for text-to-image models. In *ICCV*, 2023. 1, 2, 5, 7

[35] Michael J Wilber, Chen Fang, Hailin Jin, Aaron Hertzmann, John Collomosse, and Serge Belongie. BAM! the behance

artistic media dataset for recognition beyond photography. In *ICCV*, 2017. 6

[36] Yuguang Yao, Xiao Guo, Vishal Asnani, Yifan Gong, Jiancheng Liu, Xue Lin, Xiaoming Liu, and Sijia Liu. Reverse engineering of deceptions on machine- and human-centric attacks. *Foundations and Trends in Privacy and Security*, 2024. 3

[37] Chin-Yuan Yeh, Hsi-Wen Chen, Shang-Lun Tsai, and Sheng-De Wang. Disrupting image-translation-based deepfake algorithms with adversarial attacks. In *WACVW*, 2020. 2

[38] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5

[39] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Ngai-Man Cheung, and Min Lin. A recipe for watermarking diffusion models. *arXiv preprint arXiv:2303.10137*, 2023. 3