electronics

MDPI

Article

Temporal Adaptive Attention Map Guidance for Text-to-Image Diffusion Models

Sunghoon Jung [1] and Yong Seok Heo [1,2,*]

[1] Department of Artificial Intelligence, Ajou University, Suwon 16499, Republic of Korea; tamtam211@ajou.ac.kr
[2] Department of Electrical and Computer Engineering, Ajou University, Suwon 16499, Republic of Korea
* Correspondence: ysheo@ajou.ac.kr

Abstract: Text-to-image generation aims to create visually compelling images aligned with input prompts, but challenges such as subject mixing and subject neglect, often caused by semantic leakage during the generation process, remain, particularly in multi-subject scenarios. To mitigate this, existing methods optimize attention maps in diffusion models, using static loss functions at each time step, often leading to suboptimal results due to insufficient consideration of varying characteristics across diffusion stages. To address this problem, we propose a novel framework that adaptively guides the attention maps by dividing the diffusion process into four intervals: initial, layout, shape, and refinement. We adaptively optimize attention maps using interval-specific strategies and a dynamic loss function. Additionally, we introduce a seed filtering method based on the self-attention map analysis to detect and address the semantic leakage by restarting the generation process with new noise seeds when necessary. Extensive experiments on various datasets demonstrate that our method achieves significant improvements in generating images aligned with input prompts, outperforming previous approaches both quantitatively and qualitatively.

Keywords: text-to-image generation; attention map-based diffusion optimization; semantic leakage

Academic Editors: Liang-Jian Deng, Jiang He and Renwei Dian

Received: 6 December 2024
Revised: 17 January 2025
Accepted: 18 January 2025
Published: 21 January 2025

Citation: Jung, S.; Heo, Y.S. Temporal Adaptive Attention Map Guidance for Text-to-Image Diffusion Models. *Electronics* **2025**, *14*, 412. https://doi.org/10.3390/electronics14030412

1. Introduction

Text-to-image generation aims at creating images based on a given text prompt, with the core objective of understanding the meaning of the text and visually representing it through high-quality and diverse images. Recently, this technology has made significant advancements [1–3] in tandem with developments in diffusion model-based generative methods [4,5] and large-scale language models [6,7]. Because of its versatility, it is now being applied across various industries, including art, education, and marketing, to enhance creativity, efficiency, and user experiences.

Despite the advancements, it remains challenging to accurately align the input prompt with the generated images, especially for prompts involving multiple subjects. This misalignment often results in issues such as subject neglect, subject mixing, and incorrect attribute binding [8]. Previous research attributes this problem to inaccurate attention maps in diffusion models. The nature of the attention layer in the diffusion model, which blends features to efficiently denoise the input image, often leads to semantic leakage between subjects [9].

Existing methods [8,10] try to mitigate this issue by optimizing the latents in the diffusion models through backpropagation. They intervene in the generative process of pre-trained diffusion models, using the gradient of the loss function to obtain attention maps

that accurately reflect the input prompt. Their loss is designed to maximize the attention values for each subject token while minimizing the overlap of attention maps between subjects by shifting the latent toward valid regions at each time step. However, their loss function is suboptimal as it primarily focuses on reducing the overlap between the attention maps of individual subject tokens without properly considering the corresponding areas. Furthermore, this loss function remains static across time steps, failing to adapt to the varying attributes during the denoising process of the diffusion model [8,10], which further limits its effectiveness.

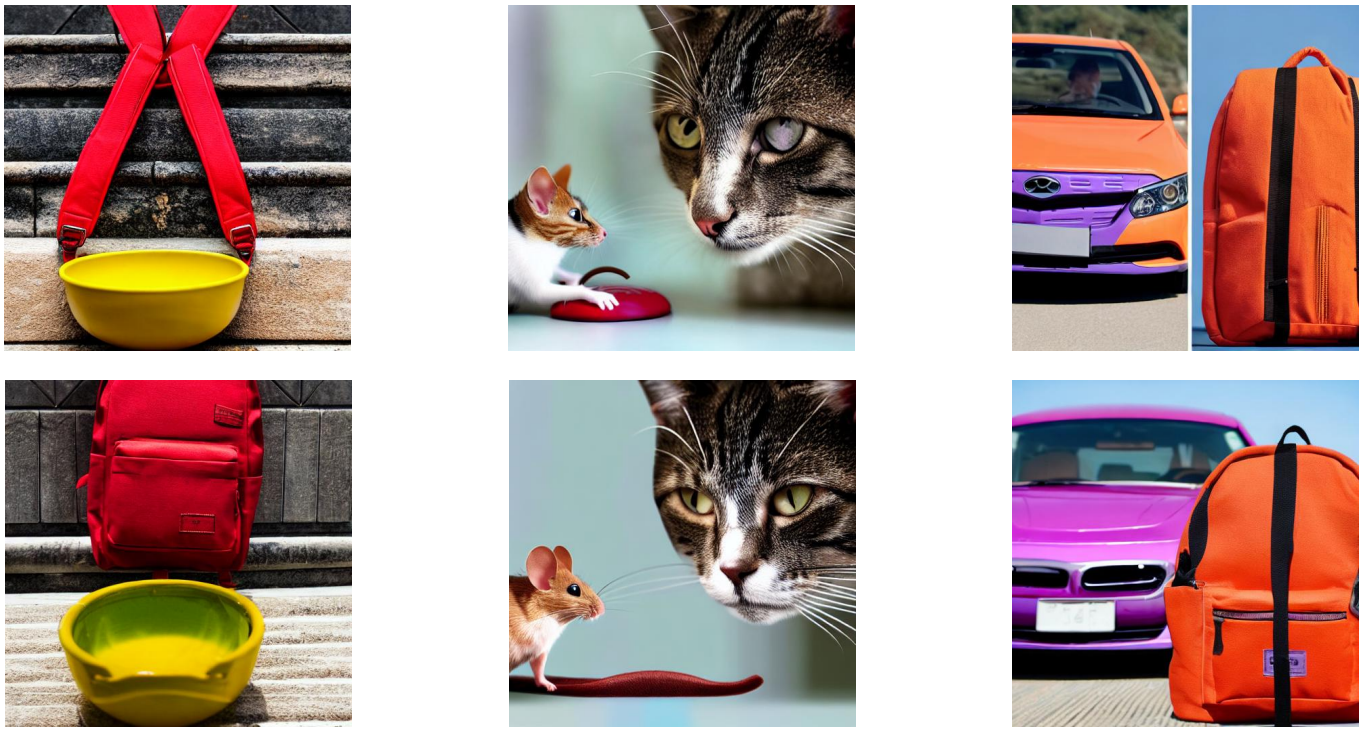
Meanwhile, other methods [9,11] address this problem by directly enforcing the attention map to match additional inputs, such as bounding boxes or segmentation maps, in order to provide an explicit layout of subjects based on the input prompt. These additional inputs are, however, often impractical and cumbersome in many scenarios.

To deal with these issues, we propose a novel attention map optimization method that utilizes a pre-trained text-to-image diffusion model. Inspired by [12], we first divide the diffusion generation steps into four intervals: initial, layout, shape, and refinement. This approach is based on the observation that the text embeddings influence different aspects at each time step, with attention maps progressively evolving to focus on the layout, object shape, and fine visual details of an object as the steps progress. We then adaptively guide the attention map for each interval using different strategies based on the proposed loss function. Specifically, to prevent overlaps between subjects and ensure sufficient space for each subject to avoid being overlooked, we propose an area loss that encourages adequate coverage for each subject. This allows our method to effectively target the specific characteristics and requirements of each interval in the diffusion process. Despite our efforts to reduce the semantic leakage in attention maps, we propose an initial seed filtering method for when it occurs. This filtering is based on the estimated mask shape using the self-attention map after the shape interval. We hypothesize that analyzing the shape of the self-attention obtained after the shape interval can provide insights into the configuration of the subjects in the final generated image. When this configuration appears to cause semantic leakage, this filtering method rejects the invalid latent, randomly selects a new seed that generates the initial noise, and restarts the generation process. This approach helps ensure the generation aligns better with the input text and produces high-quality, coherent images.

To demonstrate the superiority of the proposed method, we conducted extensive experiments on the Animal-Animal, Object-Object, and Animal-Object datasets [8]. Our approach showed significant improvements over previous methods, both quantitatively and qualitatively. Figure 1 illustrates the strengths of our approach over the recent method in [10], which suffers from semantic leakage between subjects. These issues lead to misalignment with the input text prompt, resulting in low-quality and unnatural images. In contrast, our method produces significantly more coherent and high-quality images. The contributions of the proposed method are summarized as follows:

- A temporal adaptive attention map guidance method is proposed for accurate text-to-image generation aligned with the input prompt by accounting for the characteristics of the pre-trained diffusion model at each time step.
- We divide the diffusion generation steps into four intervals: initial, layout, shape, and refinement. Leveraging the proposed loss function, we dynamically assign distinct loss terms to each interval, refining the attention maps to align the optimization process with the unique characteristics of each interval.
- An initial seed filtering method is introduced; it rejects the initial noise, randomly selects a new seed, and restarts the generation process to ensure better alignment between the generated output and the input text.

- Through various experiments, our method demonstrates significant improvements over existing approaches, showcasing its effectiveness and superiority.



"a red backpack and a yellow bowl" *"a cat and a mouse"* *"a orange backpack and a purple car"*

Figure 1. Examples of images generated using input prompts. The first row shows results of InitNO [10], while the second row presents results of the proposed method. Each column of results from recent works shows issues related to semantic leakage, subject neglect (**first column**), subject mixing (**second column**), and incorrect attribute binding (**third column**), whereas our method generates improved results.

2. Related Works

Text-to-image generation is a challenging problem in computer vision. To address this issue, numerous studies have employed various generative approaches. Early works predominantly utilized models based on generative adversarial networks (GANs), including multi-stage generation approaches [13–18] and the contrastive learning method [19]. Additionally, auto-regressive approaches were also explored [20–23]. Recently, diffusion models [4,5] have emerged as the leading paradigm, producing outstanding results. The integration of large-scale vision-language models [2,3,23,24] has enabled transformative developments in the text-to-image domain. However, producing results that accurately adhere to the given text prompt remains a challenge, even with the ability to synthesize high-quality images.

To overcome this issue, some works enhance network design and integrate large language models to offer more precise text embedding guidance. Balaji et al. [25] trained an ensemble of expert denoisers, each specialized for a specific generation stage, to enhance diffusion model capacity. Ramesh et al. [1] leveraged CLIP latent space by converting text embeddings into image embeddings, enabling better capture of textual information from text in the generated images. Saharia et al. [3] leveraged a pre-trained large language model to accurately capture textual information and enhanced the diffusion process to generate high-resolution, photorealistic images with detailed textures. Segalis et al. [26] trained text-to-image models with captions relabeled by a captioning model to improve dataset quality. Xue et al. [27] expanded the network architecture with a mixture of experts for more accurate

textual information capture. However, such approaches necessitate the development of new text-to-image models, making them incompatible with widely adopted existing models.

Alternatively, another direction of work has been explored: training-free strategies, consistent with our approach. Liu et al. [28] generate images for complex text description, such as a combination of objects, object relations, and human facial attributes, by utilizing multiple diffusion models, each responsible for different concepts, and combining their outputs. With structuring the input prompt by parsing techniques such as constituency trees, Feng et al. [29] process the cross-attention layers to ensure proper alignment between the tokens and the outputs. However, these methods exhibit issues of semantic leakage, similar to those observed in their pre-trained text-to-image generation model, Stable Diffusion [2]. On the other hand, Chefer et al. [8] introduced a latent optimization scheme, named Generative Semantic Nursing (GSN), which adjusts the latent code during the denoising process to better integrate semantic information from the text. Agarwal et al. [30] improved the latent optimization objectives in GSN to address key issues in cross-attention maps, specifically attention overlap and attention decay. Similarly, Li et al. [31] enhanced the objectives by designing a loss function to ensure that the attribute token aligns with the desired object region while also reducing the total variation in the attention maps. Recently, Guo et al. [10] proposed updating the initial noise, which balances under-optimization and over-optimization, leveraging the well-known distribution of the initial latent. This approach helps the optimized latent remain in the valid region, preserving the attention maps' properties for preventing semantic leakage. However, their latent optimization strategies are suboptimal, as they fail to adequately account for the nature of the diffusion generation process, which exhibits varying characteristics across different diffusion stages. As a result, the generated outputs exhibit issues of semantic leakage. To address these limitations, we propose an adaptive optimization pipeline inspired by the observations in [12].

3. Preliminary

3.1. Stable Diffusion Model

Our study is based on the Stable Diffusion model (SD) [2], a family of latent diffusion models. SD performs denoising in the latent space of an autoencoder. Before training SD, the autoencoder itself must be trained to establish a latent space that maps an input $I \in \mathcal{X}$ to a latent code z, and vice versa. The latent space of the autoencoder is defined by its components as follows:

$$\begin{aligned} z &= \varepsilon(I), \\ I &\cong \mathcal{D}(z), \end{aligned} \tag{1}$$

where the encoder $\varepsilon(I)$ encodes the input I into a latent representation z, and the decoder $\mathcal{D}(z)$ reconstructs the input x from its latent z. This latent space enables SD to efficiently operate in a compressed and structured representation, significantly reducing computational complexity while maintaining high-quality results.

When the training of SD begins, the model learns the distribution of noise added to the input latent z_t at each time step t, following the process of denoising diffusion probabilistic modeling (DDPM) [5]. This enables the model to progressively denoise the latent representation and reconstruct the target output.

Furthermore, diffusion models can be trained with additional conditions, known as conditional models. SD is one such conditional diffusion model and typically utilizes text embeddings generated by a pre-trained CLIP text encoder [32] as conditioning inputs. For a given text prompt $\mathcal{P}$, the corresponding text embeddings are represented as $c = CLIP(\mathcal{P})$, where c serves as the conditioning information to guide the diffusion process, enabling SD to produce outputs that align closely with the provided text prompt. This integration

of text embeddings with the diffusion model forms the foundation for SD's text-to-image generation capability.

The SD model is trained to denoise the noisy latent z_t at an arbitrary time t. The training of the SD model uses the following loss function L, which is designed to minimize the reconstruction error between the predicted and true noise during the denoising process:

$$L = \mathbb{E}_{z\sim\varepsilon(I),c,\epsilon\sim\mathcal{N}(0,1),t}\|\epsilon - \epsilon_\theta(z_t, c, t)\|_2^2, \tag{2}$$

where ϵ is the true noise which is added to clean latent z, and $\epsilon_\theta(\cdot)$ is the predicted noise from the model parameterized with θ.

The generation process begins by sampling an initial latent noise $z_T \sim \mathcal{N}(0,1)$ from a standard Gaussian distribution. The SD model then progressively denoises this latent until the cleaned latent z_0 is obtained. Finally, z_0 is passed through the decoder $\mathcal{D}(\cdot)$ to generate an image conditioned on the text prompt $\mathcal{P}$.

3.2. Attention Layer

Cross-attention layer. The alignment between input text and the resulting image in SD stems from the cross-attention layer. An input text prompt $\mathcal{P} = \{p_1, p_2, ... p_L\}$ is sequentially encoded into text embeddings by the CLIP text encoder [32], producing text embeddings $c = CLIP(\mathcal{P})$. The text embedding c is projected into the key $\mathbf{K}$ and the value $\mathbf{V}$ for the attention mechanism. On the other hand, the query $\mathbf{Q}$ is projected from the intermediate features obtained by forwarding the latent through the diffusion network [2]. A cross-attention map $\mathbf{A}^c \in \mathbb{R}^{hw\times L}$ is computed as

$$\mathbf{A}^c = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}), \tag{3}$$

where d is a channel dimension of the feature maps, L is the maximum length of text embeddings, and hw represents the spatial dimension of the intermediate features obtained from the latent. For simplicity, the time step index t and layer index l of the attention map are omitted.

Self-attention layer. Unlike the cross-attention layer, the query, key, and value are projected from intermediate features obtained by the latent. The self-attention layer generates globally coherent structures by self-attention maps $\mathbf{A}^s \in \mathbb{R}^{hw\times hw}$ that show correlations between the global spatial region and each image token.

4. Proposed Method

The aim of our proposed method is to generate a precisely aligned image based on the input text prompt $\mathcal{P}$, without the need for extra training or additional controls. First, we extract the text embedding c from prompt $\mathcal{P}$ using the CLIP [32] text encoder, where $c = CLIP(\mathcal{P})$. Then, we extract the indices of a specific group of words from the text embedding c. Here, S represents a set of indices of target subjects in the text embedding. Additionally, U denotes a set of indices that include embeddings for attributes and special tokens (e.g., <sot>, <eot>), where attributes refer to words describing target subjects, such as color or numerical attributes. By leveraging those index sets, such as S and U, our approach adaptively adjusts the attention map through a combination of latent optimization and direct manipulation of the attention map at each time step during the generative diffusion process.

As discussed in [12], the denoising process has the coarse-to-fine nature. First, the rough layout of the subject is established. Then, the coarse appearance of each subject takes shape. Finally, fine details are refined to enhance visual quality. Building on this characteristic, we propose temporal adaptive objectives that allow attention maps to dynamically incorporate temporal semantic information. This approach facilitates a smooth transition through the stages of rough layout formation, shape construction, and the addition of fine visual details. As shown in Figure 2, this is achieved by dividing the diffusion denoising process into four distinct intervals: initial, layout, shape, and refinement.

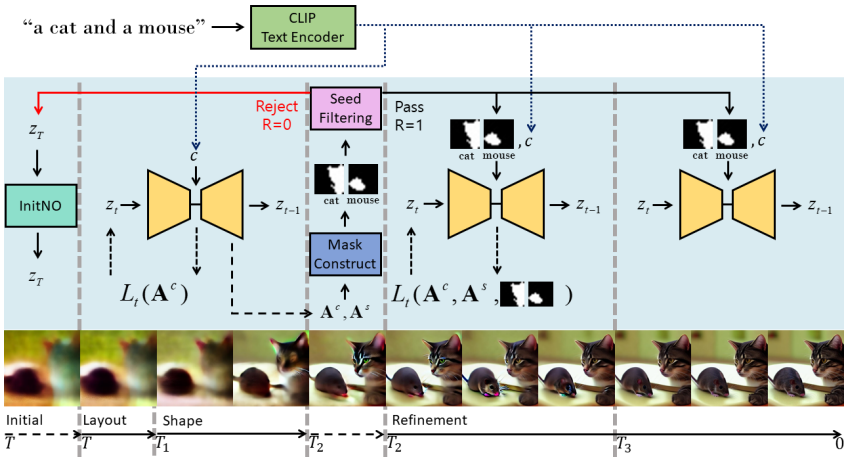


Figure 2. Overview of the proposed method. We divide the denoising sampling process of the diffusion model into four intervals: initial, layout, shape, and refinement. For each interval, we define an objective function that considers the dynamically changing influence of the text embedding over time and optimize the latent accordingly. In the final time step of the shape interval, the subject's shape within the latent is determined; we construct a mask of each subject. We then validate whether the latent is valid using seed filtering. If the latent is deemed invalid, we return to the initial interval, resample new Gaussian noise, and repeat the process. If the latent is validated as valid, we perform latent optimization and attention map blocking using the constructed mask. After T_3, only attention map blocking is applied.

At the initial interval, at time $t = T$, the latent is optimized following [10] to move it into the valid regions, so that the generated image aligns well with the input prompt. However, optimization at this interval alone is insufficient to generate an image that is fully faithful to the prompt due to the feature mixing characteristic inherent in the denoising diffusion process [9]. Therefore, the subsequent optimization procedure is carefully designed to address these challenges and leverage the inherent and unique characteristics of the generative diffusion process at each time step.

During the layout interval, optimization is performed to define the position of each subject in the cross-attention map by maximizing pixel activation values. The optimized results help ensure that each subject is appropriately activated.

In the shape interval, our optimization focuses on shaping the coarse appearance of each subject. The optimization updates the latent to help prevent the cross-attention maps of each subject from overlapping and maintain distinct areas. As a result of this optimization, the subject can form their appearance within appropriate, non-overlapping regions. In the final step of the shape interval, the coarse shape of each subject can be estimated using the cross- and self-attention maps, resulting in a binary mask for each subject [12]. Using these masks, a seed filtering method is introduced to evaluate the latent representation. If the latent is likely to lead to semantic leakage, the current process is discarded and the optimization restarts. Otherwise, the process proceeds seamlessly.

In the refinement interval, our optimization focuses on forming the fine visual details of each subject. The goal is to prevent overlap between different subjects while refining the coarse appearances established in earlier intervals, ensuring that details are accurately shaped within each subject's designated area. To achieve this, we control the shape of each subject by a dual optimization scheme, combining backward guidance through latent optimization and forward guidance through direct manipulation of the attention map. Unlike [9], which relies on external layout information, our method utilizes the estimated mask information for the dual guidance. The detailed procedure for each interval is explained in the following subsections.

4.1. Layout Interval (T, T_1)

Within the interval $t \in (T, T_1)$, text embedding influences the layout of each subject in the generated images. Similar to [8,10], we aim to ensure that at least one patch in the cross-attention map exhibits a high activation value for a specific subject. This helps to emphasize the subject's prominence at its designated position, and to achieve this, the corresponding loss L_{act} is defined by

$$L_{act} = 1 - \min_{i \in S} \max_{x} \mathbf{A}^{c}_{avg}(x, i), \tag{4}$$

where i is the index of the target subject and x is the row-wise pixel coordinate of the cross-attention map. $\mathbf{A}^{c}_{avg} \in \mathbb{R}^{hw \times L}$ represents the averaged cross-attention maps aggregated from the cross-attention layers, where L is the maximum length of text embedding c, and hw represents the total number of pixels in the feature map. $\mathbf{A}^{c}_{avg}(x, i)$ denotes a probability value of the row-wise pixel coordinate x and token index i, which indicates correlation between the x-th pixel and i-th text token embedding. Therefore, the objective for the layout interval is defined using Equation (4) as follows:

$$L_t = L_{act}, \quad t \in (T, T_1). \tag{5}$$

Then, the latent z_t in this interval is updated as follows:

$$z_t \leftarrow z_t - \eta_t \nabla_{z_t} L_t, \tag{6}$$

where η_t is a hyper-parameter for the update step size at each time step t.

4.2. Shape Interval $[T_1, T_2)$

Within the shape interval $t \in [T_1, T_2)$, a more refined shape for each subject is formed. During this interval, feature blending between subjects often occurs, leading to subject mixing and even subject neglect when one of the subject areas is small.

In order to prevent this problem, it is essential to ensure sufficient area for each subject to avoid being overlooked. To achieve this, we propose an area loss, L_{area}, that promotes adequate coverage for each subject, defined by

$$L_{area} = \sum_{i,j \in S, \forall i<j} \frac{1}{N} \frac{\sum_{x} \min(\mathbf{A}^{c}_{avg}(x, i), \mathbf{A}^{c}_{avg}(x, j))}{\min(\sum_{x} \mathbf{A}^{c}_{avg}(x, i), \sum_{x} \mathbf{A}^{c}_{avg}(x, j))}, \tag{7}$$

where i and j are indices representing different target subjects, and N denotes the total number of such paired subjects. x represents a row-wise pixel coordinate of the cross-attention map. In Equation (7), the numerator represents the region of overlap between different subjects, while the denominator corresponds to the area of the subject with the smaller activated region within each subject pair. The gradient derived from this area loss

helps prevent latent overlap, promoting attention maps that ensure adequate coverage for each object. Similar to Equation (4), averaged cross-attention is used for computing the loss function. Therefore, the objective function for optimizing the latent z_t in this interval is defined as follows:

$$L_t = L_{area}, \quad t \in [T_1, T_2). \tag{8}$$

Then, the latent z_t is updated in a similar manner by Equation (6).

4.3. Seed Filtering

The shape information for each subject token is approximately determined in the time step $t = T_2$, which marks the end of the shape interval. Despite latent optimization up to T_2, there is no guarantee that the latent variable is placed within a valid region to generate a faithful image aligned with the prompt. Therefore, we propose a shape information-based latent filtering algorithm to determine whether the current latent is valid or not. Unlike manual filtering, which requires reviewing the generated results after the generation is complete, the proposed seed filtering automatically validates the latent during intermediate stages. This enables the generation process to restart early if necessary, significantly enhancing time efficiency, particularly in large-scale image generation tasks, while providing a more practical and scalable solution.

As described in [12], the shape information can be represented in a binary mask format by clustering self-attention maps, with labels determined based on overlap scores between the clusters and cross-attention maps. Specifically, we first aggregate the self-attention maps from the self-attention layers, in a manner similar to the aggregation process of cross-attention maps, as defined in Equation (4). The aggregated self-attention map, originally with dimensions $hw \times hw$, is reshaped to $h \times w \times hw$, so that each pixel is represented by hw channels, corresponding to the total number of pixels in the feature map. This reshaped map is then clustered using the K-Means algorithm, resulting in a segment map labeled with K distinct labels, as exemplified in Figure 3. Subsequently, the segment map is used to generate a binary mask $\mathbf{C}_k \in \mathbb{R}^{hw \times 1}$ for each label $k \in \{1, \ldots, K\}$, assigning a value of 1 to regions with the label k and 0 to all other regions. Each $\mathbf{C}_k$ is assigned to the target subject indexed by $i \in S$, based on a score $s(k, i)$, defined as follows:

$$s(k, i) = \frac{\sum_x (\mathbf{C}_k(x) \mathbf{A}^c_{avg}(x, i))}{\sum_x \mathbf{C}_k(x)}, \tag{9}$$

where $\mathbf{A}^c_{avg}$ represents the aggregated cross-attention map. Specifically, $\mathbf{C}_k$ is assigned to the label $argmax_i\, s(k, i)$ if the maximum score satisfies the threshold condition (i.e., $max_i\, s(k, i) > \sigma$); otherwise, $\mathbf{C}_k$ is labeled as background. After labeling all $\mathbf{C}_k$, those assigned to the same target subject index i are combined to create the corresponding subject mask $\mathcal{M}_i \in \mathbb{R}^{hw \times 1}$. Consequently, a set of masks of target subjects, $\mathcal{M} = \{\mathcal{M}_i\}_{i \in S}$, is constructed.

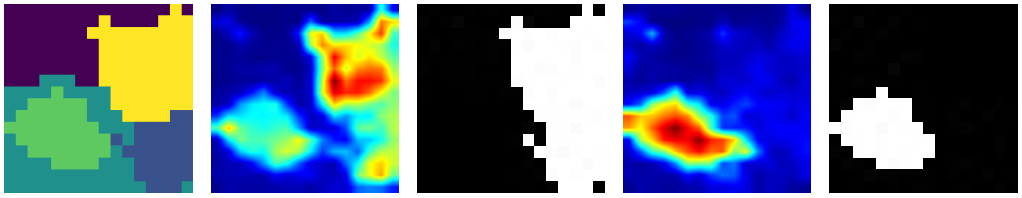
The mask vectors for different subjects are orthogonal to each other, meaning that the inner product between them is zero. Figure 3 exemplifies the estimated mask for each subject. These masks approximate the shapes of the subject tokens. Since there is no ground-truth reference shape available for evaluating the mask in our scenario, we instead use the number of clusters in the mask as a substitute metric. Specifically, if the number of clusters for each subject exceeds the desired number, it indicates that the text embedding's attention has extended to unintended regions, potentially causing semantic leakage. Based on this insight, we define a Boolean variable R_i for the i-th subject as follows:

$$R_i = \begin{cases} 1, & if\ CCA(\mathcal{M}_i) < N_i \\ 0, & otherwise \end{cases}, \tag{10}$$

where $CCA(\mathcal{M}_i)$ represents the number of connected components for the i-th subject, determined using the connected component analysis method [33]. N_i is the desired number of clusters for the i-th subject, which is manually derived from user input. Alternatively, this process can be automated using a large language model (LLM) from the input prompt. Then, a Boolean variable R is defined as follows:

$$R = R_1 \wedge \cdots \wedge R_N, \tag{11}$$

where $\wedge$ represents a logical AND operation. When $R = 1$, the latent possesses the desired shape structure for each subject and proceeds to the next step, allowing the denoising sampling to continue. Otherwise, the current latent shows potential for semantic leakage, and the entire denoising procedure restarts with a new initial noise seed. The restart count has a maximum limit R_{max}, which is set to prevent infinite filtering loops.



(a) (b) (c) (d) (e)

Figure 3. Visualization of mask construction. This figure illustrates the mask construction results for the prompt "*a cat and a mouse*". (**a**) presents a segment map obtained by clustering self-attention map at $t = T_2$. (**b**,**c**) depict the cross-attention map and the constructed mask for the text token "*cat*", respectively. Similarly, (**d**,**e**) show the cross-attention map and the constructed mask for the text token "*mouse*", respectively.

4.4. Refinement Interval, $[T_2, 0]$

In this interval, fine visual details are added, while the overall shape of each subject is further refined. To effectively refine the fine details of each subject and prevent semantic leakage, each subject needs to remain contained within its defined shape, avoiding any overlap with the regions of other subjects. To achieve this, we fully leverage the shape information of each subject as a mask, employing a dual optimization that sequentially combines latent optimization and attention map blocking using the mask, similar to [9]. The latent optimization is used to obtain a more reliable latent vector, effectively preventing the leakage of fine details between the subject and the background. Meanwhile, the attention map blocking prevents the leakage of fine details between subjects by directly masking the attention maps. Unlike [9], where the layout information is provided by the user, our method automatically derives the mask of each subject from the self-attention maps.

Attention map blocking. Attention map blocking directly modifies the attention maps using the mask $\mathcal{M}$ during the refinement interval, as the network processes latent inputs in a forward pass. The blocking method varies depending on the type of attention map. The cross-attention maps for each subject token in S are blocked in regions corresponding to other subject tokens in S to prevent overlap. Additionally, the cross-attention maps of tokens included in U are not blocked, while the cross-attention maps of tokens not included in either S or U are entirely blocked. For self-attention maps, blocking is applied to the pixels corresponding to each subject through the mask $\mathcal{M}$, ensuring that the regions for different subject do not overlap.

Specifically, the attention map blocking begins with generating a background mask, which corresponds to the regions excluding the target subject specified in the prompt. The background mask $\mathcal{M}_{BG} \in \mathbb{R}^{hw \times 1}$, which excludes any occupied subject areas, is defined as follows:

$$\mathcal{M}_{BG}(x) = \begin{cases} 0 & \text{if } \mathcal{M}_i(x) = 1 \text{ for any } i \in S, \\ 1 & \text{otherwise,} \end{cases} \tag{12}$$

where $\mathcal{M}_i(x)$ represents a mask value of subject mask $\mathcal{M}_i$ at pixel coordinate x.

The blocking mask $\mathbf{M}'_i \in \mathbb{R}^{hw \times 1}$ for the target subject, representing the area occupied by either the i-th target's subject or background, is defined as follows:

$$\mathbf{M}'_i(x) = \begin{cases} \mathbf{v_0} & \text{if } \mathcal{M}_i(x) = 1 \vee \mathcal{M}_{BG}(x) = 1, \\ \mathbf{v}_{-\infty} & \text{otherwise,} \end{cases} \tag{13}$$

where $\vee$ denotes the logical OR operation. $\mathbf{v_0}$ and $\mathbf{v}_{-\infty}$ represent vectors consisting entirely of the values of zero and $-\infty$, respectively.

Then, a cross-attention block map $\mathbf{M}_c = \left[\mathbf{m}_c^{(1)}, \ldots, \mathbf{m}_c^{(L)}\right] \in \mathbb{R}^{wh \times L}$ for the cross-attention layer is defined by

$$\mathbf{m}_c^{(i)} = \begin{cases} \mathbf{M}'_i & \text{if } i \in S \\ \mathbf{v_0} & \text{else if } i \in U \\ \mathbf{v}_{-\infty} & \text{otherwise,} \end{cases} \tag{14}$$

where $\mathbf{m}_c^{(i)}$ is the i-th column vector in $\mathbf{M}_c$.

This block map $\mathbf{M}_c$ ensures that each target subject does not encroach on the regions of other subjects. Simultaneously, meaningful attributes contributing to image generation remain unblocked, while irrelevant textual components are effectively filtered out.

Similarly, a self-attention block map $\mathbf{M}_s = \left[\mathbf{m}_s^{(1)}, \ldots, \mathbf{m}_s^{(wh)}\right] \in \mathbb{R}^{wh \times wh}$ is defined by

$$\mathbf{m}_s^{(i)} = \begin{cases} \mathbf{M'}_{G(i)} & \text{if } G(i) \neq 0 \\ \mathbf{v_0} & \text{otherwise,} \end{cases} \tag{15}$$

where the cluster index $G(x)$ of each pixel x is defined as follows:

$$G(x) = \begin{cases} i & \text{if } \mathcal{M}_i(x) = 1,\ i \in S, \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

Unlike the cross-attention block maps, the self-attention block maps focus solely on blocking interference between different subjects. The cross-attention and self-attention block maps are illustrated in Figure 4.

Then, attention map blocking is defined as

$$\hat{\mathbf{A}} = \text{softmax}(\mathbf{Q}\mathbf{K}^T + \mathbf{M}), \tag{17}$$

where $\hat{\mathbf{A}}$ represents the blocked cross-attention map, with $\hat{\mathbf{A}}^c$ denoting the blocked cross-attention map for cross-attention layers and $\hat{\mathbf{A}}^s$ denoting the blocked self-attention map for self-attention layers, respectively. Similarly, the block map $\mathbf{M}$ denotes $\mathbf{M}_c$ for cross-attention

layers and $\mathbf{M}_s$ for self-attention layers, respectively. Attention map blocking is applied not only during latent optimization but also throughout the denoising process.

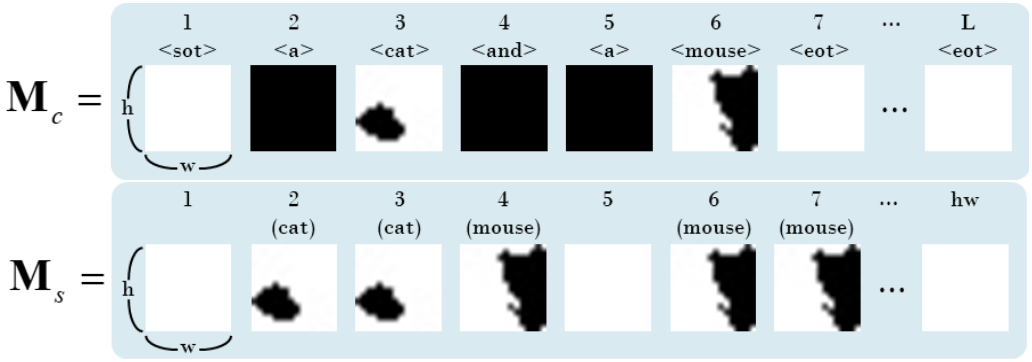


Figure 4. Visualization of attention map blocking mask. This figure shows cross-attention block map $\mathbf{M}_c$ and self-attention block map $\mathbf{M}_s$ for prompt "*a cat and a mouse*". Each column vector in the block maps $\mathbf{M}_c$ and $\mathbf{M}_s$, represented as $\mathbf{m}_c$ and $\mathbf{m}_s$, respectively, is reshaped to $\mathbb{R}^{h\times w}$. The regions shown in black in the figure have a value of $-\infty$, while the white regions have a value of 0. For $\mathbf{M}_c$, column indices range from 1 to the maximum token length L and are enclosed in "<>" to denote the corresponding text token below each index. For $\mathbf{M}_s$, column indices range from 1 to the maximum spatial coordinates hw, with indices corresponding to a subject's region represented in "()".

Latent optimization. During the refinement interval, latent optimization aims to ensure that each subject's attention map is shaped within the mask $\mathcal{M}_i$. Accordingly, we design a loss function $L_{\text{cross},i}$ to guide the cross-attention map of the i-th target subject to form within the desired region, similar to [9], as follows:

$$L_{\text{cross},i} = 1 - \frac{\sum_x \mathbb{1}\{\mathcal{M}_i(x)=1\}\hat{\mathbf{A}}^c_{avg}(x,i)}{\sum_x \mathbb{1}\{\mathcal{M}_i(x)=1\}\hat{\mathbf{A}}^c_{avg}(x,i) + \alpha\sum_x \mathbb{1}\{\mathcal{M}_i(x)=0\}\hat{\mathbf{A}}^c_{avg}(x,i)}, \tag{18}$$

where $\mathbb{1}\{\cdot\}$ represents an indicator function that returns one when the condition in the brace is met and zero otherwise, and α is the number of target subjects in the prompt, similar to [9]. $\hat{\mathbf{A}}^c_{avg}$ represents the mean cross-attention map, obtained by averaging the blocked cross-attention map $\hat{\mathbf{A}}^c$ across heads and layers. In Equation (18), the first term in the denominator represents the aggregation of values from the cross-attention map within the i-th cluster, while the second term aggregates values outside the i-th cluster. By setting the weighting factor α higher than one, it further penalizes the leakage of cross-attention values outside the cluster, encouraging concentration within the same cluster.

Similarly, a loss function $L_{\text{self},i}$ is defined to guide the self-attention map of the i-th target subject as follows:

$$L_{\text{self},i} = 1 - \frac{\sum_x \sum_{y\in\mathbf{y}_i} \mathbb{1}\{G(x)=i\}\hat{\mathbf{A}}^s_{avg}(x,y)}{\sum_x \sum_{y\in\mathbf{y}_i} \mathbb{1}\{G(x)=i\}\hat{\mathbf{A}}^s_{avg}(x,y) + \alpha\sum_x \sum_{y\in\mathbf{y}_i} \mathbb{1}\{G(x)\neq i\}\hat{\mathbf{A}}^s_{avg}(x,y)}, \tag{19}$$

where $\mathbf{y}_i$ is a pixel coordinate set that is placed in a cluster labeled i, that is, $\mathbf{y_i} = \{y|G(y)=i, y\in\{1,...,hw\}\}$, and the same α as in Equation (18) is used. Similarly, in Equation (19), the first term in the denominator aggregates values from the self-attention map within the i-th cluster, while the second term gathers values outside the i-th cluster. By setting the weighting factor α above one, it enhances the penalty on the dispersion of self-attention values outside the cluster, thereby promoting focus within the designated

cluster. $\hat{\mathbf{A}}^s_{avg}$ represents the mean self-attention map, obtained by averaging the blocked self-attention map $\hat{\mathbf{A}}^s$ across heads and layers. Then, a mask loss L_{mask} is defined by

$$L_{mask} = \sum_{i \in S} (L^2_{\text{cross},i} + L^2_{\text{self},i}). \tag{20}$$

The objective function for optimizing the latent z_t in this interval is defined as follows:

$$L_t = L_{mask}, \quad t \in [T_2, T_3). \tag{21}$$

Subsequently, the latent z_t in this interval is updated in a manner similar to Equation (6) for $t \in [T_2, T_3)$. After the latent update, during the denoising process, the latent z_t is denoised to z_{t-1} with attention map blocking applied again. In the interval $[T_3, 0]$, further latent optimization is omitted due to its minimal impact, while denoising with attention map blocking continues.

The entire algorithm of the proposed method is described in Algorithm 1.

Algorithm 1 Full algorithm

Input: pre-trained text-to-image generation diffusion model $SD(\cdot)$ and Decoder $Dec(\cdot)$, input prompt $\mathcal{P}$
Output: generated image I

Extract text embedding from [32], $c = CLIP(\mathcal{P})$
Sample a random Gaussian noise $z_T \sim N(0, 1)$
$z_T \leftarrow$ initial noise optimization (z_T, T, c) from [10]
$R_{iter} = 0$
while $R_{iter} < R_{max}$ **do**
  for $t = T$ to 0 **do**
    if $t \in [T, T_2)$ **then**
      _, $\mathbf{A}^s$, $\mathbf{A}^c \leftarrow SD(z_t, c, t)$
      if layout interval, $t \in [T, T_1)$ **then**
        $L_t = L_{act}(\mathbf{A^c})$ in Equation (4)
      else if shape interval, $t \in [T_1, T_2)$ **then**
        $L_t = L_{area}(\mathbf{A^c})$ in Equation (7)
      end if
    else if refinement interval, $t \in [T_2, T_3)$ **then**
      _, $\mathbf{A}^s$, $\mathbf{A}^c \leftarrow SD(z_t, c, t, \mathcal{M})$
      $L_t = L_{mask}(\mathbf{A}^s, \mathbf{A}^c, \mathcal{M})$ in Equation (20)
    end if
    $z_t \leftarrow z_t - \eta \nabla_{z_t} L_t$
    if $t = T_2$ **then**
      _, $\mathbf{A}^s$, $\mathbf{A}^c \leftarrow SD(z_t, c, t)$
      $\mathcal{M} \leftarrow MaskConstruction(\mathbf{A}^s, \mathbf{A}^c)$ from [12]
      $R \leftarrow SeedFiltering(\mathcal{M})$
      if $R = 0$ **then**
        $R_{iter} = R_{iter} + 1$
        go to line 2
      end if
    end if
    if $t \in [T, T_2)$ **then**
      $z_{t-1} \leftarrow SD(z_t, c, t)$
    else
      $z_{t-1} \leftarrow SD(z_t, c, t, \mathcal{M})$
    end if
  end for
  $R_{iter} = R_{max}$
end while
$I \leftarrow Dec(z_0)$
return I

5. Experiments

5.1. Experimental Settings

Implementation Details. We utilize the official Stable Diffusion (SD) v1.4 [2] for the text-to-image generation model with a guide scale of 7.5, following [10]. The time intervals for the diffusion denoising process are established [12] as follows: The initial time step T is set to $T = 50$. The layout interval range is $[T, T_1)$, where T_1 is set to 45. Following this, the shape interval range is $[T_1, T_2)$, where T_2 is set to 35. Finally, during the refinement interval, the range in which latent optimization is applied spans $[T_2, T_3)$, where T_3 is set to 15. We optimize a latent variable using attention layers from both the encoder block and decoder blocks of a U-Net transformer, following [8]. For pre-processing, the attention maps are smoothed using a Gaussian filter with a kernel size of 3 and a standard deviation of 0.5, similar to [8]. To construct a mask, we follow the settings of [12] for the same attention layer for optimization. The attention maps are utilized in layers 8 to 11, and 14 to 19 in SD for both attention map blocking and optimization. The update step size η_t in Equation (6) is uniformly divided between 1.0 and 0.5 across the number of sampling steps, and these values are applied at each time step in decreasing order, starting from 1.0 and ending at 0.5. For seed filtering, we set $K = 5$ for K-means clustering on the self-attention map, and the threshold σ is set at $\sigma = 0.3$ for subject labeling for each cluster, following [12]. The token identification of the target subject is manually derived from user input; however, it can also be automated using a large language model (LLM) [7]. Attribute indices U in the prompt are identified using a part-of-speech (POS) tagger, which extracts key descriptive elements. Additionally, special tokens are assigned based on the length of the prompt to manage and structure the length of the input prompt effectively.

Datasets. We utilize openly available datasets [8,31] to compare the performance of text-based image generation models. In this work, the input prompts contain two subjects categorized into two types: animals and objects. The datasets from [8] consist of three: "Animal-Animal", "Object-Object", and "Animal-Object". Specifically, each dataset has a distinct format: "an [animal A] and an [animal B]" for Animal-Animal, "a [color A] [object A] and a [color B] [object B]" for Object-Object, and "an [animal] and/with a [color][object]" for Animal-Object. The prompts are created using 12 animal types, 12 object types, and 11 colors. Both the Animal-Animal and Object-Object subjects include 66 paired prompts, while the Animal-Object subject comprises 144 paired prompts. Additionally, the "Color-Objects-Scene" dataset [31] uses the format "a [color A] [animal or object A] and a [color B] [animal or object B] [scene]". This dataset consists of 60 prompts and is generated using three animal types, nine object types, nine colors, and six scene types.

Metrics. For quantitative evaluations of text-to-image generation methods, we use CLIP [32] similarities, following the protocol established in [8]. Specifically, we use three types of similarities: *Full Prompt Similarity*, *Minimum Object Similarity*, and *Text-Text Similarity*. For each prompt, we generate 64 images for evaluations by using randomly sampled seeds.

Full Prompt Similarity represents the average CLIP cosine similarity between the feature obtained from the full text prompt and the features from a set of 64 generated images. However, *Full Prompt Similarity* alone may not capture certain semantic issues, such as the neglect of the subject within the image. To address this, we also evaluate the *Minimum Object Similarity*, which accounts for cases where specific subjects may be underrepresented or overlooked. To compute the *Minimum Object Similarity*, we first split the input prompt into two sub-prompts, each containing a single subject (e.g., "animalA" or "objectA"). We then calculate the CLIP cosine similarities for each sub-prompt with the generated images, taking the lower of the two scores as the *Minimum Object Similarity*. However, due to the modality gap between CLIP's image and text embedding spaces [34,35], mixed subject representations and semantic leakage may still occur, as observed in [8]. To address

this, and following the approach in [8], we compute *Text-Text Similarity* using CLIP. This leverages CLIP's robust semantic framework to ensure that all subjects and attributes from the original prompt are accurately represented and prioritized. To evaluate this similarity, we first generate captions for each image using the BLIP image-captioning model [36]. Then, we calculate the average *Text-Text Similarity* between the original prompt and the generated captions, aggregating the results across images produced with the same prompt but different random seeds. This process is repeated for each subset, and the final results are averaged across all prompts within the subset.

To further evaluate, we utilize the GenEval score from [37]. The GenEval is an object-focused framework designed to evaluate compositional image properties such as object co-occurrence, position, count, and color. To calculate the GenEval score, each generated image is processed using object detection and instance segmentation with the Mask2Former [38], trained on MS COCO, provided by the MMDetection toolbox [39]. Detected objects are checked against the target subjects specified in the prompt, and compositional image properties are assessed for the identified objects. The user defines these properties for each prompt in advance, and evaluation methods vary dependent on each property, as specified in [37].

5.2. Quantitative Comparison

Tables 1–3 present the quantitative comparison results of the proposed method against previous approaches. For brevity, we denote the *Full Prompt Similarity* and *Minimum Object Similarity* as Full and Min, respectively, and the *Text-Text Similarity* as Text. In the tables, we also present the relative decrease in each metric (in percentage) compared to the proposed method.

Table 1. Quantitative comparison on the Animal-Animal dataset.

	Full (↑)	**Min (↑)**	**Text (↑)**
SD [2]	0.3164 (−7.93%)	0.2205 (−14.88%)	0.7675 (−8.07%)
AnE [8]	0.3386 (−1.46%)	0.2537 (−2.09%)	0.8086 (−3.15%)
DnB [31]	0.3336 (−2.92%)	0.2472 (−4.59%)	0.8065 (−3.39%)
InitNO [10]	0.3420 (−0.48%)	0.2563 (−1.07%)	0.8233 (−1.39%)
Ours	0.3437	0.2591	0.8349

Table 2. Quantitative comparison on the Object-Object dataset.

	Full (↑)	**Min (↑)**	**Text (↑)**
SD [2]	0.3359 (−8.58%)	0.2376 (−13.62%)	0.7641 (−7.63%)
AnE [8]	0.3634 (−1.11%)	0.2727 (−0.82%)	0.8153 (−1.44%)
DnB [31]	0.3564 (−3.01%)	0.2649 (−3.67%)	0.8068 (−2.47%)
InitNO [10]	0.3668 (−0.19%)	0.2735 (−0.55%)	0.8227 (−0.55%)
Ours	0.3675	0.2750	0.8272

Table 3. Quantitative comparison on the Animal-Object dataset.

	Full (↑)	**Min (↑)**	**Text (↑)**
SD [2]	0.3451 (−4.74%)	0.2493 (−8.22%)	0.7924 (−5.77%)
AnE [8]	0.3606 (−0.47%)	0.2707 (−0.33%)	0.8344 (−0.78%)
DnB [31]	0.3525 (−2.71%)	0.2638 (−2.88%)	0.8312 (−1.16%)
InitNO [10]	0.3630 (0.20%)	0.2719 (0.09%)	0.8414 (0.06%)
Ours	0.3623	0.2716	0.8409

The Animal-Animal dataset. Because of the semantic similarity between animal subjects, the Animal-Animal dataset is more prone to semantic leakage compared to other datasets. Table 1 presents the quantitative comparison results on the Animal-Animal dataset, where the proposed method outperforms all others across all metrics. In particular, compared to InitNO [10], the proposed method achieves improvements of 0.48% in *Full Prompt Similarity*, 1.07% in *Minimum Object Similarity*, and 1.39% in *Text-Text Similarity*. The improvement in *Full Prompt Similarity* demonstrates that the proposed method generates images that are more faithful to the overall content of the input prompt. Furthermore, the improvements in *Minimum Object Similarity* and *Text-Text Similarity* demonstrate the effectiveness of the proposed method in addressing semantic leakage.

The Object-Object dataset. While the semantic similarity between subjects in the Object-Object dataset is not as pronounced as in the Animal-Animal dataset, it nonetheless exhibits factors that heighten the risk of semantic leakage. Additionally, each prompt has specified the colors of the objects, introducing challenges related to attribute binding. From Table 2, the proposed method outperforms all others across all metrics. Compared to InitNO [10], the proposed method achieves improvements of 0.19% in *Full Prompt Similarity*, 0.55% in *Minimum Object Similarity*, and 0.55% in *Text-Text Similarity*. Similar to the results on the Animal-Animal dataset, the proposed method, leveraging the temporal adaptive attention optimization pipeline, effectively resolves issues of semantic leakage. Moreover, for the Object-Object dataset, the proposed method effectively addresses attribute binding issues, as evidenced by the results in *Text-Text Similarity*.

The Animal-Object dataset. The Animal-Object dataset consists of subjects from different classes, resulting in relatively low semantic similarity between subjects compared to other datasets. Additionally, the object subjects in the Animal-Object dataset include an attribute for color, similar to the Object-Object dataset, which assigns a color attribute to each object. Table 3 shows that the proposed method achieved the second-best performance compared to previous methods. While the proposed method closely approaches the performance of InitNO, with slight differences of 0.20% in *Full Prompt Similarity*, 0.09% in *Minimum Object Similarity*, and 0.06% in *Text-Text Similarity*, it demonstrates comparable effectiveness to InitNO in addressing semantic leakage, particularly within the Animal-Object dataset.

The Color-Objects-Scene dataset. Unlike other datasets, the Color-Objects-Scene dataset includes prompts that allow combinations of all classes of animals and objects. It also specifies the color attributes of each subject and adds a postfix describing the scene or scenario, making it the dataset with the highest prompt complexity. Table 4 shows that the proposed method outperforms all others across all metrics in this dataset. Compared to InitNO [10], the proposed method achieves improvements of 0.31% in *Full Prompt Similarity*, 0.90% in *Minimum Object Similarity*, and 0.66% in *Text-Text Similarity*. These results demonstrate that the proposed method effectively prevents semantic leakage and generates images that adhere more faithfully to the prompt, even for more complex prompts.

Table 4. Quantitative comparison on the Color-Objects-Scene dataset.

	Full (↑)	**Min (↑)**	**Text (↑)**
SD [2]	0.3641 (−3.22%)	0.2363 (−12.72%)	0.7033 (−5.01%)
AnE [8]	0.3730 (−0.85%)	0.2676 (−1.18%)	0.7247 (−2.12%)
DnB [31]	0.3679 (−2.21%)	0.2619 (−3.26%)	0.7245 (−2.14%)
InitNO [10]	0.3751 (−0.31%)	0.2683 (−0.90%)	0.7354 (−0.66%)
Ours	0.3762	0.2708	0.7404

To further evaluate the proposed method, we employ the GenEval score with attribute binding as the compositional image property. The GenEval score evaluates compositional image properties based on object detection, inherently reflecting how well the target subjects are represented in the image. As shown in Table 5, the proposed method outperforms the existing works. These results indicate that the proposed method effectively achieves faithful representation of subjects in the prompt, which in turn contributes to addressing attribute binding issues.

Table 5. Quantitative comparison results on the Color-Objects-Scene dataset with the GenEval score [37].

	SD [2]	**AnE [8]**	**DnB [31]**	**InitNO [10]**	**Ours**
GenEval score (↑)	0.110 (−0.72%)	0.301 (−0.24%)	0.281 (−0.29%)	0.372 (−0.07%)	0.398

Generation time. Table 6 presents the average generation time for each method. To evaluate the generation time of each method, 100 images with a resolution of 512 × 512 pixels were randomly generated for each prompt in the Color-Objects-Scene dataset using a single NVIDIA RTX 6000 Ada Generation GPU with 50-step sampling, and the results were averaged. According to Table 6, SD generates a single image in approximately 8.63 s, while AnE and DnB require 9.07 and 17.77 s, respectively. InitNO takes 21.51 s, and our proposed method requires a similar time of 22.08 s. Specifically, InitNO involves iterative latent optimizations at specific time steps, namely, $t = 50, 40, 30$. For these time steps, the latent is optimized until the loss falls below the threshold or the iteration number reaches the predefined maximum. Similarly, our method also performs iterative latent optimization at $t = 50$, but it does not conduct such optimization at $t = 40$ or $t = 30$. Instead, it utilizes seed filtering to reject unsuitable noise and resamples initial noise to restart the image generation process. The results from Table 6 indicate that the average generation time in our method, which uses the repeated generation process by seed filtering, is comparable to the iterative latent refinement process in InitNO.

Table 6. Comparison of generation time for various methods.

	SD [2]	**AnE [8]**	**DnB [31]**	**InitNO [10]**	**Ours**
Average time (s)	8.63	9.07	17.77	21.51	22.08

5.3. Qualitative Comparison

Animal-Animal Dataset. Figure 5 shows qualitative comparison results from the Animal-Animal dataset, where most methods struggle with semantic leakage due to the high semantic similarity between subjects. This semantic leakage often leads to issues such as subject neglect or subject mixing in the generated image. For example, in the results for "*a cat and a mouse*" shown in the first and second rows in Figure 5, previous methods fail to address the semantic leakage. Both SD and AnE exhibit subject mixing in the first row, producing a mouse with cat-like features. For example, the shape of the mouse's mouth and its feet resemble those of a cat. DnB does not exhibit subject mixing; however, it fails to adhere to the prompt, producing a cat, a computer mouse, and an incomplete representation of a mouse. In the second row, they exhibit subject neglect: SD and DnB generate two cats instead of a cat and a mouse, while AnE produces a cat with mouse-like features. Although InitNO considers subject separation, it still fails to preserve distinct features for each subject. Semantic leakage with InitNO is evident in two examples: in the first row, it produces an imperfect cat with the size of a mouse, and in the second row, it generates a hybrid cat characterized by a

cat-like face, mouse ears, and mouse feet. Conversely, the proposed method achieves well-separated representations of a cat and a mouse with a plausible layout. This demonstrates the critical importance of encouraging spatially specific areas through the used of estimated mask information for the attention map to preserve distinct subject features and achieve clear semantic separation. By effectively focusing on such regions, the proposed method better resolves subject-related issues and enhances overall image quality.

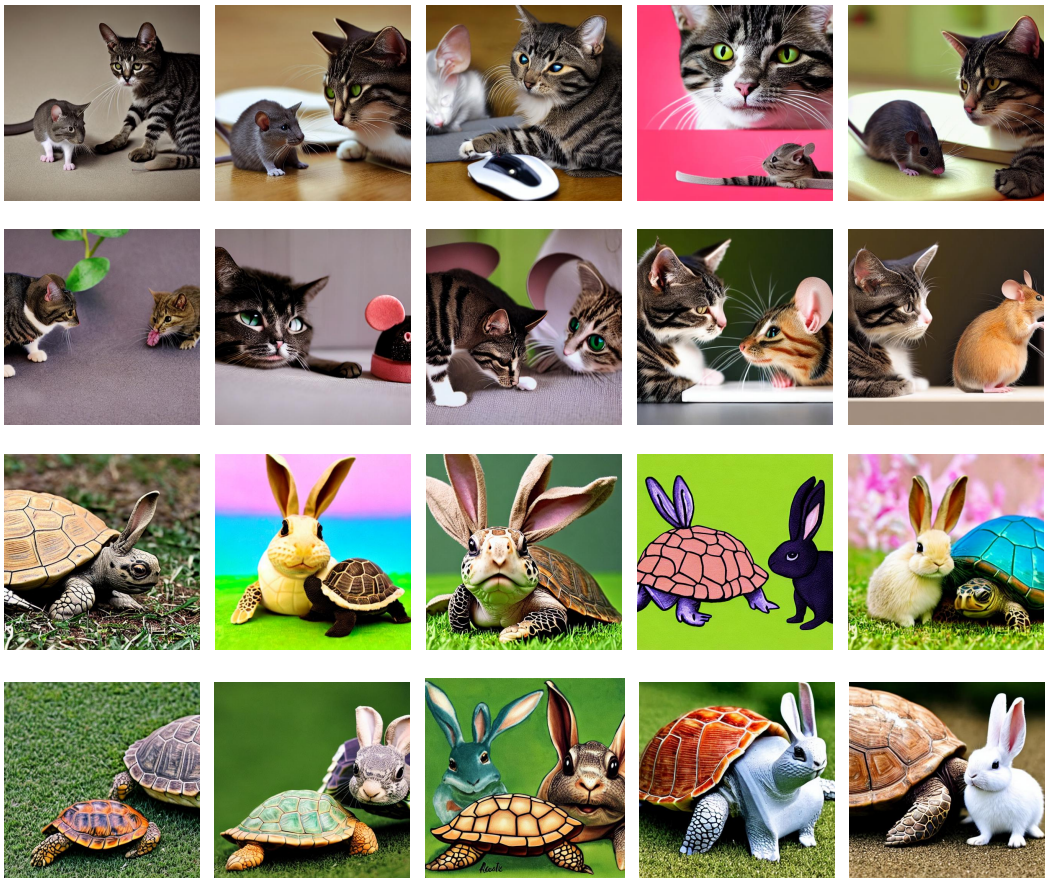


Figure 5. Qualitative comparison results on the Animal-Animal dataset. From left to right, the columns represent the results of Stable Diffusion (SD) [2], Attend-and-Excite (AnE) [8], Divide-and-Bind (DnB) [31], Initial Noise Optimization (InitNO) [10], and the proposed method (ours). Each row shows images resulting from the same input prompt and a random seed. The first and second rows correspond to the prompt "*a cat and a mouse*", while the third and fourth rows show results for the prompt "*a turtle and a rabbit*".

Similarly, the semantic leakages of previous method observed in the "*a turtle and a rabbit*" become even more pronounced, as shown in the third and fourth rows in Figure 5. Both SD and AnE continue to struggle with semantic leakage, including subject mixing and neglect. For instance, SD and DnB generate turtles with rabbit ears, or they produce either two turtles or two turtle–rabbit hybrids. AnE often produces a rabbit head on a turtle's body, failing to maintain proper subject separation. While InitNO occasionally avoids subject mixing, the resulting images often exhibit cartoonish textures, indicating they are out of distribution compared to the real dataset. Additionally, the turtle shell sometimes includes rabbit ears, implying that the latent representation is positioned in an invalid region. In contrast, the proposed method generates well-separated and authentic images. This demonstrates that the proposed adaptive optimization effectively ensures the latent space is positioned within the desired valid regions.

Object-Object Dataset. Since the Object-Object dataset contains the highest frequency of attribute words in the prompts, it naturally presents more challenges related to attribute binding and semantic leakage. Figure 6 shows a qualitative comparison of results from this dataset. Previous works suffer from semantic leakage, leading to issues such as subject neglect and subject mixing. For instance, in the prompt "*a white car and a black bowl*", as shown in the first and second rows of Figure 6, both SD and DnB struggle with subject neglect. SD generates only a white car, while DnB produces white and black bowls instead. Similarly, the images produced by AnE exhibit not only subject neglect but also evidence of subject mixing. For example, in the second row of Figure 6, AnE produces a white car that is blended with a black bowl. InitNO, while often able to generate separate depictions of the two objects, fails to faithfully align with the given prompts. For instance, in the second row of Figure 6, InitNO's depiction of a black bowl shows two bowls, one white and the other black, which deviates from the prompt's intent.

Attribute binding also presents noticeable challenges. In the case of "*a yellow backpack and a purple chair*", previous works struggle to correctly bind attributes. For instance, SD and AnE generate a purple backpack and a yellow chair in the fourth row of Figure 6, while DnB generates a black backpack, a purple chair, and a yellow chair. AnE also produces an unnatural layout of the chair and backpack. The results of InitNO also suffer from incorrect attribute binding. In the third and fourth rows of Figure 6, InitNO generates a yellow backpack and a yellow chair from both resulting images. In contrast, the proposed method successfully generates results where the attributes specified in the prompt are correctly bound to the multiple subjects, and the subjects themselves are accurately represented without any subject neglect or subject mixing. This is clearly demonstrated in the fourth column of Figure 6, where the white car and black bowl, as well as the yellow backpack and purple chair, are distinctly represented with the attributes as specified in the prompt. Thus, the results presented in Figure 6 demonstrate that the proposed method effectively addresses subject neglect and subject mixing. Furthermore, it outperforms in attribute binding by leveraging temporal attention map optimization.

Animal-Object Dataset. Figure 7 shows a comparison of results from the Animal-Object dataset. It is observed that most previous works successfully separate subjects; however, they often push the latent variable outside the trained data distribution, resulting in a noticeable decline in image quality. For instance, in the case of "*a bear and a purple bowl*", not only does SD often struggle, but AnE, DnB, and InitNO also frequently produce outputs with cartoonish textures, which indicates an issue with latent variables operating out of the trained data distribution. Even though AnE and InitNO can avoid cartoonish texture to some extent, they still face significant challenges in aligning the generated images with the input prompts.

A similar issue arises with "*a turtle and a blue chair*", where most previous methods struggle with semantic accuracy. SD fails to produce semantically aligned outputs, often generating only a blue chair without the turtle. Even when two subjects are generated, the outputs frequently exhibit cartoonish textures. Similarly, DnB occasionally produces outputs with cartoonish texture. While AnE and InitNO produce well-separated subjects with realistic textures, they still fail to align with the input prompts, for instance, generating two turtles instead of a single turtle and a chair. As shown in the third and fourth rows of Figure 7, both AnE and InitNO produce two blue turtles and a chair-like object. Additionally, AnE fails to accurately bind the color attribute. Even when both methods succeed in generating well-aligned images, the quality of the resulting image is not plausible. AnE fails to produce a complete and coherent chair, while InitNO struggles to generate a complete form of a turtle. In contrast, our method creates a more natural scene layout

while preserving authentic textures. This improvement can be attributed to the proposed attention map optimization, which ensures both texture quality and semantic accuracy.

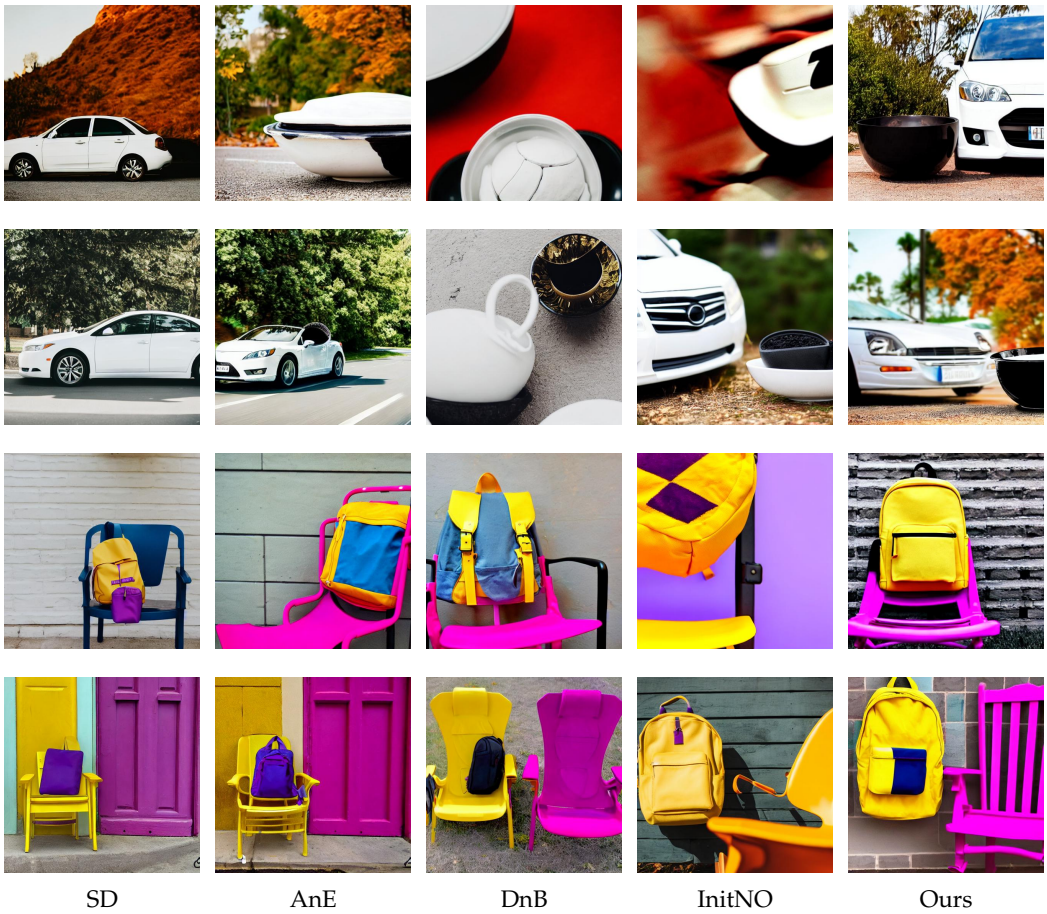


Figure 6. Qualitative comparison results on the Object-Object dataset. From left to right, the columns represent the results of Stable Diffusion (SD) [2], Attend-and-Excite (AnE) [8], Divide-and-Bind (DnB) [31], Initial Noise Optimization (InitNO) [10], and the proposed method (ours). Each row shows images resulting from the same input prompt and a random seed. The first and second rows correspond to the prompt "*a white car and a black bowl*", while the third and fourth rows show results for the prompt "*a yellow backpack and a purple chair*".

Color-Objects-Scene Dataset. Figure 8 shows a qualitative comparison of results from the Color-Objects-Scene dataset. Due to the complexity of its prompts, which surpasses those of other datasets, previous works have encountered issues with both semantic leakage and inadequate scene generation. For example, in the results for "*a black cat and a red suitcase in the library*" in the first row of Figure 8, most previous works fail to comprehensively generate both the subjects and the scene. SD fails to generate the suitcase, while AnE, DnB, and InitNO fail to depict the library. Furthermore, all previous methods show cartoonish textures in their resulting images. A similar issue arises with "*a blue bird and a brown backpack on the street, rainy driving scene*" in the second row of Figure 8, where AnE and DnB fail to generate the scene and their resulting images have cartoonish textures. While SD and InitNO successfully generate the target subjects and the scene, their results remain incomplete: SD produces a blue bird without eyes, and InitNO generates a mixture combining features of a blue bird and a human. In contrast, the proposed method successfully generates the target subjects with accurately bound color attributes for both prompts, achieving precise scene generation. These results demonstrate that the proposed method maintains high semantic accuracy even for complex prompts.

Figure 7. Qualitative comparison results on the Animal-Object dataset. From left to right, the columns represent the results of Stable Diffusion (SD) [2], Attend-and-Excite (AnE) [8], Divide-and-Bind (DnB) [31], Initial Noise Optimization (InitNO) [10], and the proposed method (ours). Each row shows images resulting from the same input prompt and a random seed. The first and second rows correspond to the prompt "*a bear and a purple bowl*", while the third and fourth rows show results for the prompt "*a turtle and a blue chair*".

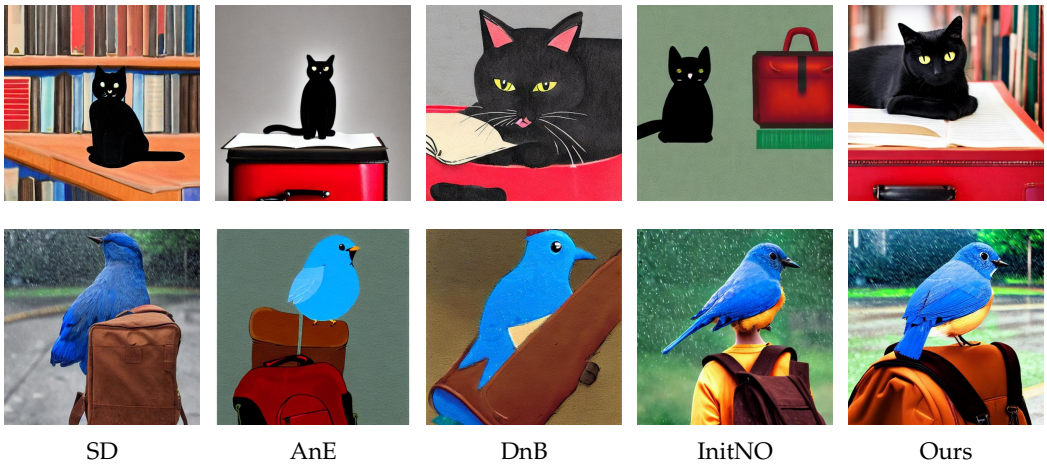


Figure 8. Qualitative comparison results on the Color-Objects-Scene dataset. From left to right, the columns represent the results of Stable Diffusion (SD) [2], Attend-and-Excite (AnE) [8], Divide-and-Bind (DnB) [31], Initial Noise Optimization (InitNO) [10], and the proposed method (ours). Each row shows images resulting from the same input prompt and a random seed. The first row shows results for the prompt "*a black cat and a red suitcase in the library*", while the second row corresponds to the prompt "*a blue bird and a brown backpack on the street, rainy driving scene*".

5.4. Ablation Study

5.4.1. Ablation Study on Components

To investigate the effects of the components in the proposed method, such as area loss in Equation (7), dual optimization in Equations (17) and (20), and seed filtering in Equations (10) and (11), we conducted ablation experiments by removing one component at a time to evaluate its individual contribution. Table 7 presents the results of the ablation study. All experiments were conducted on the Animal-Animal dataset [8] using 64 different seed numbers for a single prompt, with CLIP similarity scores used for evaluation.

Table 7. Effects of components of the proposed method. Relative decreased rate compared to M1 is shown in parentheses.

Method	Seed Filtering	L_{area}	Dual Optimization	Full (↑)	Min (↑)	Text (↑)
M1	✓	✓	✓	0.3434	0.2591	0.8340
M2		✓	✓	0.3421 (−0.38%)	0.2582 (−0.35%)	0.8316 (−0.29%)
M3	✓		✓	0.3426 (−0.23%)	0.2583 (−0.31%)	0.8326 (−0.17%)
M4	✓	✓		0.3442 (0.23%)	0.2593 (0.08%)	0.8308 (−0.38%)

From Table 7, the proposed method is used as the baseline and is referred to as M1 in the table. M2 represents a version of the method that excludes seed filtering. M3 removes area loss L_{area} during the shape interval and uses L_{act} instead for latent updates, where the objectives of M3 at each time step are defined as

$$L_t = \begin{cases} L_{act} & \text{if } t \in [T, T_2) \\ L_{mask} & \text{else if } t \in [T_2, T_3). \end{cases} \tag{22}$$

In M4, dual optimization is excluded during the refinement interval. The latent optimization objective L_{mask} is replaced with the activation loss L_{act}, and attention map blocking is not applied. Specifically, the objective functions of M4 for each time step are defined as

$$L_t = \begin{cases} L_{act} & \text{if } t \in [T, T_1) \\ L_{area} & \text{else if } t \in [T_1, T_2) \\ L_{act} & \text{else if } t \in [T_2, T_3). \end{cases} \tag{23}$$

First, we validate the effectiveness of the seed filtering by comparing M1 and M2. In Table 7, by adding seed filtering to M2, M1 shows improvements, with a 0.38% increase in *Full Prompt Similarity*, a 0.35% increase in *Minimum Object Similarity*, and a 0.29% increase in *Text-Text Similarity* compared to M2. These results demonstrate that seed filtering effectively identifies and mitigates problematic latents that cause semantic leakage. Figure 9 illustrates generated images and their corresponding subject masks from different seed numbers, highlighting cases that are expected to be filtered out by the seed filtering. In the third column of Figure 9, the number of clusters for the rabbit subject exceeds the desired number of rabbits, leading to semantic leakage in the generated images. For instance, in the first and second rows in Figure 9, the generated images show subject mixing, such as a cat with rabbit ears. Similarly, in the second and third rows, multiple rabbits are generated in the images, which is an unintended outcome. These artifacts highlight the challenges posed by invalid latents, which are difficult to address solely through optimization. Therefore, through investigations, we demonstrate the effectiveness of seed filtering in identifying and discarding such invalid latents, thereby improving semantic accuracy.

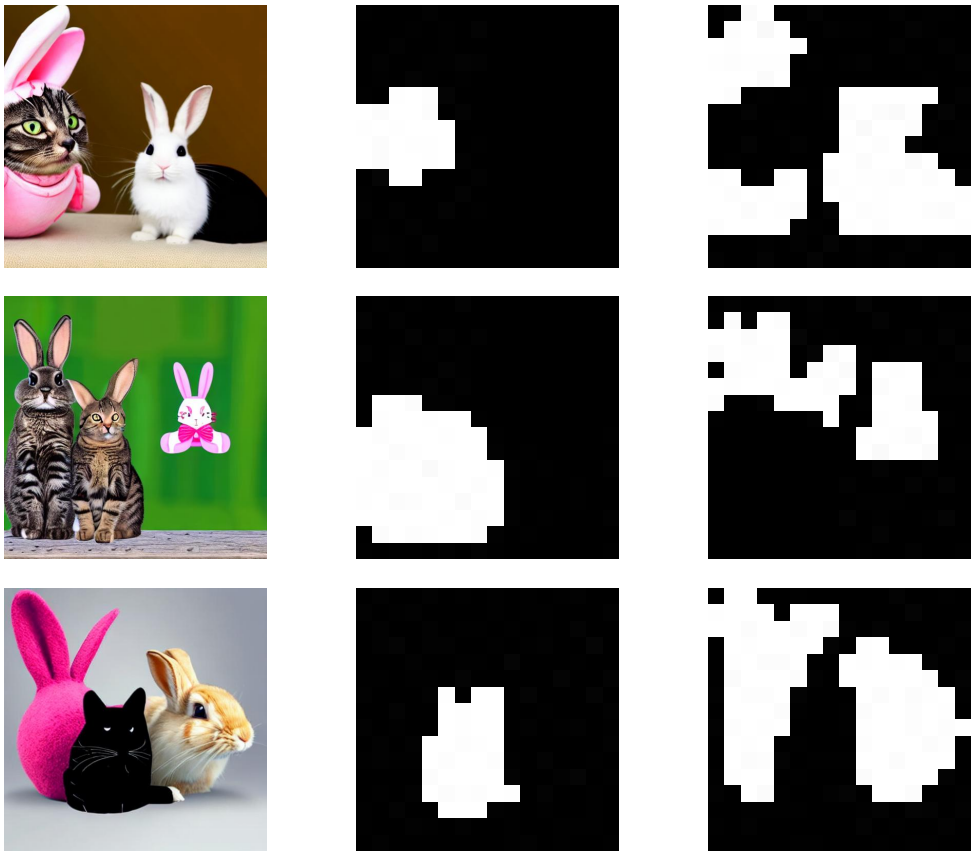


Figure 9. Examples rejected by the seed filtering. All results are generated from the prompt "*a cat and a rabbit*". From left to right, the columns display the generated images, the masks of the cat, and the masks of the rabbit.

Next, we validate the effectiveness of the area loss L_{area} by comparing M1 and M3. The area loss encourages the latent to shape separated cross-attention maps with appropriately sized regions via gradient updates. As demonstrated in Table 7, M1 outperforms M3, improving *Full Prompt Similarity* by 0.23%, *Minimum Object Similarity* by 0.31%, and *Text-Text Similarity* by 0.17%. These improvements highlight the role of area loss in mitigating semantic leakage, as further supported by the visual evidence presented in Figure 10. For instance, in the results of "*a dog and a frog*", M3 exhibits subject neglect, generating only a dog. In contrast, M1 successfully generates both a dog and a frog. Similarly, for "*a dog and a rabbit*", M3 fails to generate both subjects, while M1 successfully generates both the dog and the rabbit. Therefore, it can be concluded that preventing semantic leakage necessitates not only the separation of cross-attention maps between subjects but also the assurance that each map occupies a distinct and appropriately sized region.

Finally, by comparing M1 and M4, we investigate the dual optimization, which is combined with blocking attention maps and latent optimization using mask loss L_{mask}. The dual optimization technique is designed to focus fine visual details within the desired regions, thereby mitigating semantic leakage. As shown in Table 7, M1 improves *Text-Text Similarity* by 0.38%, but results in a decrease of 0.23% in *Full Prompt Similarity* and 0.08% in *Minimum Object Similarity*. These results demonstrate that without the dual optimization scheme during the refinement interval, prompt fidelity improves slightly, and subject neglect is marginally mitigated. However, the decrease in *Text-Text Similarity* for M4 compared to M1 indicates a more severe issue of subject mixing. Since *Text-Text Similarity* is less influenced by the discrepancies between image and text domains, it is a more appropriate metric for

assessing subject mixing, which is challenging to detect with image–text similarity-based scores. Figure 11 illustrates the subject mixing problem in M4. For the input prompt "*a cat and a bird*", M4 generates birds with a cat's head and fused cat ears, indicating issues of subject mixing. In contrast, M1 successfully generates both a complete cat and a bird. Similarly, for the input prompt "*a cat and a frog*", M4 generates frogs with a cat's body and ears, while M1 clearly separates the cat and the frog in the resulting image.

M1 M3 M1 M3

Figure 10. Qualitative comparison between M1 and M3. The first and second columns show the comparison results for "*a dog and a frog*". The third and fourth columns show the comparison results for "*a dog and a rabbit*".

M1 M4 M1 M4

Figure 11. Qualitative comparison between M1 and M4. The first and second columns show the comparison results for "*a cat and a frog*". The third and fourth columns show the comparison results for "*a cat and a bird*".

5.4.2. Ablation Study on Sampling Steps

To investigate the impact of varying the total number of sampling steps in the proposed method, we compared results with different settings. In order to reduce the sampling step from 50 to 25, we leveraged the linear nature of noise scheduling in the pre-trained text-to-image generation model Stable Diffusion by adjusting the amount of noise injection. This adjustment preserves the temporal properties of the text embedding throughout the denoising process. Consequently, the time steps T_1, T_2, and T_3, which define the boundaries of each interval, were also linearly scaled. Specifically, for a sampling of 25 steps, we adjusted T from 50 to 25, T_1 from 45 to 22, T_2 from 35 to 17, and T_3 from 15 to 7. Similar to the ablation study on the components, experiments were conducted on the Animal-Animal dataset with 64 different seed numbers per prompt. The evaluation results, using CLIP similarity scores, are displayed in Table 8.

Compared to the original 50-step sampling, the proposed method with 25-step sampling shows a slight increase, by 0.03%, in *Full Prompt Similarity* and an increase of 0.15% in

Minimum Object Similarity. However, it shows a decrease of 0.63% in *Text-Text Similarity*. The 25-step sampling method struggles to effectively mitigate semantic leakage and falls short of achieving the performance of the original 50-step sampling. This limitation is further demonstrated in Figure 12, which presents the progression of predicted images across the four stages. From the initial stage $t = T$ (first column) to the layout interval at $t = T_1$ (second column) the approximate outlines of subjects gradually emerge. As sampling advances to the shape interval at T_2 (third column), the subjects' shapes become more defined. Finally, during the refinement interval at $t = T_3$ and the final step of sampling $t = 0$, the visual details of the subjects are fully formed. As shown in Figure 12, due to insufficient latent optimization, the 25-step sampling method is more susceptible to semantic leakage compared to the original 50-step sampling approach for each prompt. For instance, the second row with "*a turtle and a rabbit*" demonstrates that the 25-step sampling fails to fully depict the turtle's head. Similarly, the fourth row with "*a cat and a monkey*", where the semantic information related to the monkey is not fully utilized, results in objects resembling toys rather than the intended monkey. In contrast, in the first and third rows in Figure 12, 50-step sampling of the proposed method generates faithful images consistent with the prompts.

Figure 12. Comparison results of different stages based on the number of sampling steps in our method. Each column shows a result from the captioned time step. For instance, the first column shows results at the initial time step of the diffusion sampling process. The first (50 steps) and second (25 steps) rows are results of "*a turtle and a rabbit*". The third (50 steps) and fourth (25 steps) rows are results of "*a cat and a monkey*". The pairs in the rows are results of different sampling steps of our method with the same initial seed.

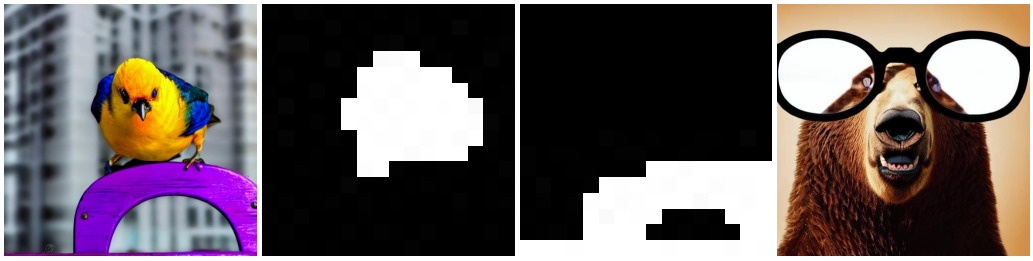
Table 8. Effects of number of sampling steps of the proposed method.

	T	T_1	T_2	T_3	Full (↑)	Min (↑)	Text (↑)
Ours (50 steps)	50	45	35	15	0.3437	0.2591	0.8349
Ours (25 steps)	25	22	17	7	0.3438	0.2595	0.8296

6. Discussion

Although the proposed method improved semantic accuracy, there are still limitations to address. First, the proposed method is limited by the expressive power of the pre-trained text-to-image generation model. If the input prompt falls outside the distribution of textual descriptions learned by the model, the results of the proposed method may not correspond to the prompt.

Second, the proposed method has a limitation in that it partially represents certain subjects when there is a significant scale difference between subjects within a natural scene (e.g., bird and bench). This limitation becomes particularly pronounced in the Animal-Object dataset, where significant scale differences between subjects within natural scenes are common. Figure 13 illustrates this problem. When given the prompt "*a bird and a purple bench*", the mask for the bird in Figure 13b has enough area to generate the entire shape of the bird, while the mask for the bench in Figure 13c only covers the area to represent part of the bench. As a result, the generated image in Figure 13a contains a fully formed bird and only a small portion of a bench. The issue of partial subject generation due to scale differences stems from the area loss in the proposed method. This loss prioritizes expanding smaller subjects in natural scenes but does not account for scale differences, leading to a reduction in the activated regions for larger subjects. As a result, larger objects tend to be partially represented. To address this, future work should focus on developing an improved area loss that ensures balanced representation for both smaller and larger subjects within natural scenes.



(a) (b) (c) (d)

Figure 13. Examples of the limitations of the proposed method. (**a**) shows a resultant image for the prompt "*a bird and a purple bench*". (**b**,**c**) show masks for a bird and a bench, respectively. (**d**) shows a resulting image for the prompt "*a bear with glasses*".

Third, the proposed method addresses semantic leakage by utilizing binary masks to prevent overlap between target subjects on the attention map. However, this approach may not be suitable for complex scenarios in which different subjects exhibit intricate interactions. For instance, when a prompt includes a transparent object, such as glasses, it may fail to accurately capture the characteristics of such objects. Figure 13d illustrates this example. In the resulting image, the bear appears to be wearing glasses, but no part of the bear is generated in the area where the glasses are placed. To address scenarios involving intricate interactions, one area of future work involves exploring the construction of continuous-valued masks derived from the latent space, instead of using the binary masks currently employed. These continuous-valued masks would enable the generation

of images that not only ensure distinct subject regions but also account for the interactions, including overlapping between subjects.

Fourth, the current approach statically allocates the denoising steps for each interval, which may lead to suboptimal results as the shape formation of each subject can vary depending on the prompt and initial noise. Future work could explore a dynamic partitioning approach for the time steps in the diffusion sampling process to further improve the method. This adjustment is expected to enhance the quality of the generated images.

Finally, the proposed method utilizes the ability of CLIP to represent relationships between words within a prompt. However, this is not sufficient for depicting complex textual descriptions [40]. It is necessary to more precisely reflect the relationship between modifiers and entity-nouns in the prompt. Future work could explore methods for capturing precise syntactic structures and representing them in the image domain.

7. Conclusions

In this work, we addressed the critical problem of misalignment between an input prompt and generated images in text-to-image diffusion models, particularly for complex prompts involving multiple subjects. To this end, we proposed a temporal adaptive attention map guidance method, leveraging the inherent intervals of the diffusion process: initial, layout, shape, and refinement intervals. By employing tailored optimization strategies for each interval, our approach effectively reduces semantic leakage and enhances attribute binding. Additionally, our proposed initial seed filtering method adds robustness to the generation process, ensuring better alignment with the input prompt and higher-quality outputs by rejecting invalid latents and restarting the generation when necessary. Extensive experiments on various datasets demonstrated the efficacy of our method, showcasing significant improvements over existing methods in both quantitative and qualitative evaluations.

Author Contributions: Conceptualization, S.J. and Y.S.H.; software, S.J.; validation, Y.S.H.; investigation, S.J.; writing—original draft preparation, S.J. and Y.S.H.; writing—review and editing, Y.S.H.; supervision, Y.S.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2022R1F1A1065702; and in part by the Institute of Information & communications Technology Planning & Evaluation (IITP) under the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2025-RS-2023-00255968) grant funded by the Korea government (MSIT).

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank the anonymous reviewers for their constructive comments and recommendations. During the preparation of this study, the authors used Stable Diffusion version 1.4, as described in Section 5, "Experiments", for the purposes of performance comparisons with the proposed method. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv* **2022**, arXiv:2204.06125.
2. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 10684–10695.

3. Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E.L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. Photorealistic text-to-image diffusion models with deep language understanding. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; pp. 36479–36494.
4. Dhariwal, P.; Nichol, A. Diffusion models beat gans on image synthesis. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; pp. 8780–8794.
5. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–12 December 2020; pp. 6840–6851.
6. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 5485–5551.
7. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–12 December 2020; pp. 1877–1901.
8. Chefer, H.; Alaluf, Y.; Vinker, Y.; Wolf, L.; Cohen-Or, D. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Trans. Graph. (TOG)* **2023**, *42*, 1–10. [CrossRef]
9. Dahary, O.; Patashnik, O.; Aberman, K.; Cohen-Or, D. Be yourself: Bounded attention for multi-subject text-to-image generation. In Proceedings of the European Conference on Computer Vision, Milan, Italy, 29 September–4 October 2024; pp. 432–448.
10. Guo, X.; Liu, J.; Cui, M.; Li, J.; Yang, H.; Huang, D. Initno: Boosting text-to-image diffusion models via initial noise optimization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 9380–9389.
11. Chen, M.; Laina, I.; Vedaldi, A. Training-free layout control with cross-attention guidance. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 4–8 January 2024; pp. 5331–5341.
12. Patashnik, O.; Garibi, D.; Azuri, I.; Averbuch-Elor, H.; Cohen-Or, D. Localizing object-level shape variations with text-to-image diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 22994–23004.
13. Tao, M.; Tang, H.; Wu, F.; Jing, X.Y.; Bao, B.K.; Xu, C. Df-gan: A simple and effective baseline for text-to-image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 16494–16504.
14. Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; He, X. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1316–1324.
15. Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D.N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5908–5916.
16. Zhu, M.; Pan, P.; Chen, W.; Yang, Y. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5795–5803.
17. Rani, P.; Kumar, D.; Sudhakar, N.; Prakash, D.; Shubham. Text-to-Image Synthesis using BERT Embeddings and Multi-Stage GAN. In Proceedings of the International Conference on Innovative Computing and Communications, Delhi, India, 17–18 February 2023; pp. 157–167.
18. Deng, Z.; He, X.; Peng, Y. LFR-GAN: Local Feature Refinement based Generative Adversarial Network for Text-to-Image Generation. *ACM Trans. Multimed. Comput. Commun. Appl.* **2023**, *19*, 1–18. [CrossRef]
19. Zhang, H.; Koh, J.Y.; Baldridge, J.; Lee, H.; Yang, Y. Cross-modal contrastive learning for text-to-image generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Online, 19–25 June 2021; pp. 833–842.
20. Chang, H.; Zhang, H.; Barber, J.; Maschinot, A.; Lezama, J.; Jiang, L.; Yang, M.H.; Murphy, K.P.; Freeman, W.T.; Rubinstein, M.; et al. Muse: Text-To-Image Generation via Masked Generative Transformers. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; pp. 4055–4075.
21. Ding, M.; Yang, Z.; Hong, W.; Zheng, W.; Zhou, C.; Yin, D.; Lin, J.; Zou, X.; Shao, Z.; Yang, H.; et al. Cogview: Mastering text-to-image generation via transformers. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; pp. 19822–19835.
22. Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; Sutskever, I. Zero-shot text-to-image generation. In Proceedings of the International Conference on Machine Learning, Online, 18–24 July 2021; pp. 8821–8831.
23. Yu, J.; Xu, Y.; Koh, J.Y.; Luong, T.; Baid, G.; Wang, Z.; Vasudevan, V.; Ku, A.; Yang, Y.; Ayan, B.K.; et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv* **2022**, arXiv:2206.10789.
24. Gu, J.; Zhai, S.; Zhang, Y.; Susskind, J.M.; Jaitly, N. Matryoshka diffusion models. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024; pp. 1–29.

25. Balaji, Y.; Nah, S.; Huang, X.; Vahdat, A.; Song, J.; Zhang, Q.; Kreis, K.; Aittala, M.; Aila, T.; Laine, S.; et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv* **2022**, arXiv:2211.01324.
26. Segalis, E.; Valevski, D.; Lumen, D.; Matias, Y.; Leviathan, Y. A picture is worth a thousand words: Principled recaptioning improves image generation. *arXiv* **2023**, arXiv:2310.16656.
27. Xue, Z.; Song, G.; Guo, Q.; Liu, B.; Zong, Z.; Liu, Y.; Luo, P. Raphael: Text-to-image generation via large mixture of diffusion paths. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; pp. 41693–41706.
28. Liu, N.; Li, S.; Du, Y.; Torralba, A.; Tenenbaum, J.B. Compositional visual generation with composable diffusion models. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 423–439.
29. Feng, W.; He, X.; Fu, T.J.; Jampani, V.; Akula, A.; Narayana, P.; Basu, S.; Wang, X.E.; Wang, W.Y. Training-free structured diffusion guidance for compositional text-to-image synthesis. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023; pp. 1–21.
30. Agarwal, A.; Karanam, S.; Joseph, K.; Saxena, A.; Goswami, K.; Srinivasan, B.V. A-star: Test-time attention segregation and retention for text-to-image synthesis. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 2283–2293.
31. Li, Y.; Keuper, M.; Zhang, D.; Khoreva, A. Divide & Bind Your Attention for Improved Generative Semantic Nursing. In Proceedings of the British Machine Vision Conference, Aberdeen, UK, 20–24 November 2023; pp. 1–12.
32. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning, Online, 18–24 July 2021; pp. 8748–8763.
33. Szeliski, R. *Computer Vision: Algorithms and Applications*, 2nd ed.; Springer: New York, NY, USA, 2022.
34. Liang, V.W.; Zhang, Y.; Kwon, Y.; Yeung, S.; Zou, J.Y. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; pp. 17612–17625.
35. Sheynin, S.; Ashual, O.; Polyak, A.; Singer, U.; Gafni, O.; Nachmani, E.; Taigman, Y. Knn-diffusion: Image generation via large-scale retrieval. *arXiv* **2022**, arXiv:2204.02849.
36. Li, J.; Li, D.; Xiong, C.; Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In Proceedings of the International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022; pp. 12888–12900.
37. Ghosh, D.; Hajishirzi, H.; Schmidt, L. Geneval: An object-focused framework for evaluating text-to-image alignment. In Proceedings of the Advances in Neural Information Processing Systems, New Orleans, LA, USA, 10–16 December 2023; pp. 52132–52152.
38. Cheng, B.; Misra, I.; Schwing, A.G.; Kirillov, A.; Girdhar, R. Masked-attention mask transformer for universal image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1280–1289.
39. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv* **2019**, arXiv:1906.07155.
40. Lewis, M.; Nayak, N.V.; Yu, P.; Yu, Q.; Merullo, J.; Bach, S.H.; Pavlick, E. Does clip bind concepts? probing compositionality in large image models. In Proceedings of the Findings of the Association for Computational Linguistics: EACL 2024, St. Julian's, Malta, 17–22 March 2024; pp. 1487–1500.