LogoSticker: Inserting Logos into Diffusion Models for Customized Generation

Mingkang Zhu[1] Xi Chen[2] Zhongdao Wang[3]
Hengshuang Zhao[2] Jiaya Jia[1,4]

[1]CUHK [2]HKU [3]Huawei Noah's Ark Lab [4]SmartMore

Abstract. Recent advances in text-to-image model customization have underscored the importance of integrating new concepts with a few examples. Yet, these progresses are largely confined to widely recognized subjects, which can be learned with relative ease through models' adequate shared prior knowledge. In contrast, logos, characterized by unique patterns and textual elements, are hard to establish shared knowledge within diffusion models, thus presenting a unique challenge. To bridge this gap, we introduce the task of logo insertion. Our goal is to insert logo identities into diffusion models and enable their seamless synthesis in varied contexts. We present a novel two-phase pipeline LogoSticker to tackle this task. First, we propose the actor-critic relation pre-training algorithm, which addresses the nontrivial gaps in models' understanding of the potential spatial positioning of logos and interactions with other objects. Second, we propose a decoupled identity learning algorithm, which enables precise localization and identity extraction of logos. LogoSticker can generate logos accurately and harmoniously in diverse contexts. We comprehensively validate the effectiveness of LogoSticker over customization methods and large models such as DALLE 3. Project page.

Keywords: Image Generation · Diffusion Models · Customization

1 Introduction

Recent advancements in text-to-image generative models have demonstrated unprecedented capabilities in generating high-quality images conditioned on text prompts [2, 7, 13, 17, 22, 23, 26, 28]. To allow the generation of user-specified subjects, various customization methods have been proposed to integrate user-provided concepts into these models [1, 6, 8, 14, 25]. Despite their success on everyday objects, these methods still have difficulties inserting concepts without enough commonly shared knowledge, like logos or texts. For example, Stable Diffusion [23] has adequate prior knowledge of pets. Even without customization, it can generate Corgis similar in appearance to a user-specified Corgi using only text prompts. However, it is nontrivial to synthesize precise English text using prompts [5], not to mention logos, which can additionally contain unique patterns and non-English texts.

Logos, which have extremely diversified shapes and appearances, are typically difficult to have commonalities but are crucial in applications like marketing

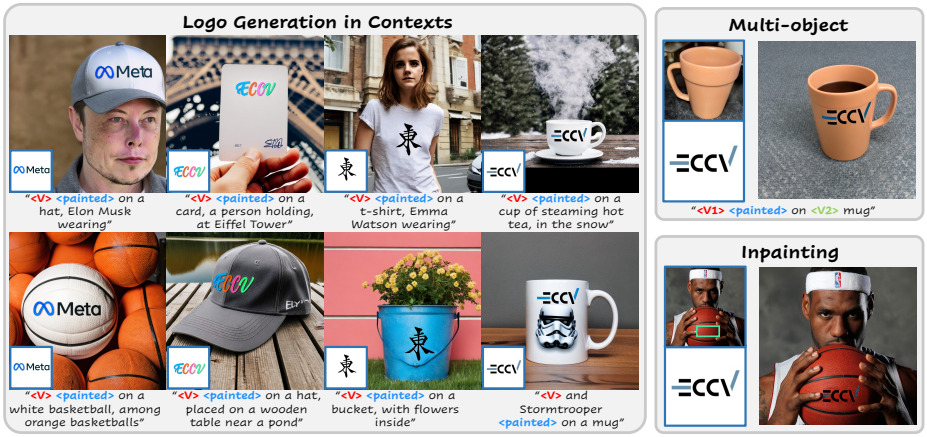

Fig. 1: Given a logo with fine-grained details, our method LogoSticker enables accurate distilling of its identity to diffusion models, thus supporting coherent text-to-image generation in diverse scenarios. It can also be extended to multi-object customization, and logos inpainting on user-given images.

and advertising. Therefore, in this work, we aim to tackle the challenge of logo insertion: given a user-provided logo to the diffusion model, our goal is to enable the model to recognize the logo, learn its identity accurately, and enable coherent generations of the logo in various scenarios.

Without adequate shared prior knowledge of diversified logos, the task of logo insertion is nontrivial. The first challenge is that diffusion models are not fully proficient in accurately conveying relationships [11], including the relation of painting things on different objects, which is essential for our task. This deficiency is more pronounced for logos, as logos are diverse and complicated and the models can hardly find references in their knowledge about where the diverse logos should appear, and how would they interact with other objects. The second challenge is the ambiguity in concept extraction from training images: It is usually not obvious which concept should be extracted from the training images. Previous methods generally rely on the diffusion model's adequate prior knowledge of commonplace items (e.g., dog or cat) [8,25], so that the class name token can attend the object they intend to learn. This approach, however, is not viable for learning diverse and complicated logos. Usual relevant class name tokens for logos generally cannot accurately attend diverse logos we aim to learn, as shown in Fig. 3. Without this ability, precise logo identity extraction is unlikely.

We propose LogoSticker to tackle this task. To address the first challenge, we need to encapsulate the relation of painting things on various objects into diffusion models. There have been attempts to represent an object-to-object relation by optimizing a text token on a few images [11]. However, we find that: the relation cannot be effectively learned and generalized with limited training data; the difficulties of learning the relation for different objects are different.

These observations lead us to introduce the Actor-Critic Relation Pre-training strategy. We curate a diverse relation dataset of various objects to paint things on. Then we fine-tune a text token and the text encoder to learn the relation. Since learning to paint things on some objects is significantly more difficult than others, we employ an actor-critic strategy to adjust the sampling probabilities of different objects. Specifically, we use a CLIP model [20] as the critic to evaluate whether the model has learned to paint things on certain objects. More samplings are given to objects that have not been mastered. In this way, the learned relation becomes stronger and more balanced among different objects.

To tackle the second challenge, we decouple the logo identity learning process into two parts: binding the target logo to a text token, and accurately learning the logo identity. Specifically, we propose to generate two distinct sets of training images for this purpose: the logo token binding set and the logo identity learning set. We generate the logo token binding set by pasting the logo on random locations of solid color backgrounds, and generate the logo identity learning set by pasting the logo on random locations of natural scenes. To make the model recognize the logo, we fix the U-Net parameters and optimize a special token on the logo token binding set so that the special text token is bound with the target logo and can attend the logo precisely. Then we fine-tune the U-Net on the logo identity learning set with the optimized special token in the prompt to accurately extract the logo identity and map it to the special text token.

With the aforementioned methods, LogoSticker can stick diverse logos in different contexts, as demonstrated in Fig. 1. We quantitatively and qualitatively evaluate LogoSticker against other customization methods. We also conduct a user study to demonstrate human evaluators' preference for our method's generations. Comparisons with state-of-the-art models including DALLE 3 [2] and text generation models like Textdiffuser-2 [4] and AnyText [33] further demonstrate LogoSticker's advantage in contextualizing logos precisely and harmoniously. We also demonstrate its versatility by showing several applications.

Our contributions can be summarized threefold: (1) We study a new problem of logo insertion, which aims to internalize user-provided logos accurately and empower consistent and diverse contextualizations. Existing customization methods largely focus on widely recognized concepts by utilizing the model's sufficient prior knowledge, while we make the exploration into complex and diverse logos. (2) We propose a novel two-phase pipeline LogoSticker for this task, consisting of the actor-critic relation pre-training algorithm followed by the decoupled identity learning algorithm. Our pipeline enables precise and seamless generations of logos in various contexts. (3) Experiment results demonstrate the effectiveness of LogoSticker over state-of-the-art customization methods and large models including DALLE 3 in logo contextualization.

2 Related Work

Text-to-image generative models. Significant progress has been made in text-to-image generative models in recent years. Taking advantage of the devel-

opment of diffusion models [10, 30–32] and large-scale cross-modal models like CLIP [20] and T5 [21], synthesizing high-fidelity and diverse images based on text descriptions has become a reality. State-of-the-art large-scale text-to-image diffusion models such as Stable Diffusion [23], Imagen [26], DALLE 2 [22], and DALLE 3 [2] have significantly advanced these capabilities. ControlNet [35] introduces more conditions to improve diffusion models' controllability. However, we find that it is still nontrivial for them to follow detailed text descriptions and precisely reconstruct user-provided logos. Our method focuses on integrating user-provided logos into these models so that they can be elicited using text prompts. The objective is to facilitate the coherent generation of these logos across varied contexts while maintaining their distinct characteristics.

Customization of generative models. Customized image synthesis focuses on implanting user-provided subjects into diffusion models and facilitating their generation in different contexts. Textual Inversion [8] tries to reconstruct user-provided subjects by optimizing a text token and adding it to the text encoder's dictionary. Dreambooth [25] uses an existing rare token and fine-tunes the model weights. Recent developments have explored customizing multiple user-specific concepts simultaneously, either by finetuning [1, 14, 15] or by zero-shot methods [6, 27]. ReVersion [11], on the other hand, tries to optimize a text token representing a general object-to-object relation. However, these methods typically focus on customizing subjects or concepts that generative models have sufficient prior knowledge of. Their performance on logos remains unsatisfactory. In contrast, our method aims to introduce complex and diverse logos, of which models have little prior knowledge, to the output domain of text-to-image diffusion models and enable their coherent generation in various contexts.

Visual text generation. Although recent text-to-image generative models can create vibrant and complex images, they often fall short of generating legible and cohesive texts. Some works claim this drawback is due to that the CLIP embedding does not contain character-level information of words in prompts which could be crucial for generating readable texts [22]. Therefore, recent methods like Imagen [26] utilize large language models like T5 [21] to achieve better text rendering quality. GlyphControl [34], Textdiffusers [4, 5], and AnyText [33] incorporate the glyph information to better condition the text generation. These methods mainly focus on English texts and cannot customize text generation. It is also nontrivial for them to follow precise descriptions of the properties or styles of the desired texts. Our method can customize text generation, which enables diffusion models to follow precise specifications of desired text properties. Our pipeline can also deal with non-English texts effortlessly.

3 Methods

Given a user-provided logo, we aim to generate the logo in different contexts precisely and cohesively. Specifically, we want the generated logo to retain its unique identity while being positioned correctly and angled suitably across diverse settings. We achieve this via a two-phase pipeline LogoSticker, in which each phase

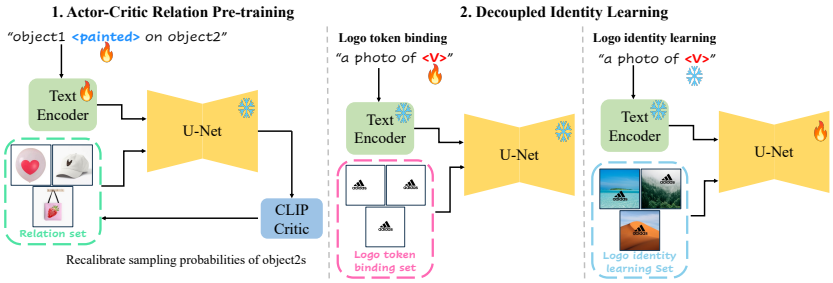

Fig. 2: The overall pipeline of our proposed LogoSticker. (1) We first pre-train the text encoder and a token <painted> in an actor-critic fashion to learn the relation of logo placement in various contexts effectively. (2) We build the logo token binding set and optimize another special token <V> to bind it with the target logo so that the target logo in training images can be localized. Then, we build a more complex logo identity learning set and fine-tune the U-Net to capture the logo identity precisely.

incorporates the relation of appearing in various contexts for logo placement and precisely learns the logo identity correspondingly. The overall pipeline is presented in Fig. 2. In this section, we first lay the groundwork by providing an overview of text-to-image diffusion models in Subsection 3.1. In Subsection 3.2, we present how we map the relation of appearing in various contexts to a text token, which in turn enables coherent logo generation in various scenes. Subsequently, we demonstrate how to make the diffusion model accurately recognize and internalize the identity of the logo in Subsection 3.3.

3.1 Preliminary

Diffusion models are a class of generative models that learn data distributions through progressively denoising Gaussian distribution samples [10, 31, 32]. Contemporary diffusion models are often executed in the compressed latent space of an autoencoder for efficiency [2, 23]. They use a pre-trained autoencoder $\mathcal{E}$ to map images x into their latent counterparts $z = \mathcal{E}(x)$. Then the latent diffusion model is trained to denoise the latent code z. The diffusion model can be conditioned on a vector $c = \gamma_\theta(P)$, where γ_θ is a pre-trained text encoder and P is the conditioning prompt. Then the latent diffusion loss can be cast as:

$$L_{LDM}(\theta) = \mathbb{E}_{z,P,t,\epsilon\sim\mathcal{N}(0,1)} \left[||\epsilon - \epsilon_\theta(z_t, t, \gamma_\theta(P))||_2^2 \right], \tag{1}$$

where t is timestep, z_t is noised latent at time t, and ϵ_θ is the denoising U-Net [24]. Our method is built on top of Stable Diffusion [23], a widely recognized and publicly accessible latent diffusion model trained on the LAION dataset [29].

3.2 Actor-Critic Relation Pre-training

As we aim for contextually appropriate and accurate logo generations in various contexts, we want the diffusion models to be capable of painting logos coher-

ently on various objects. However, diffusion models sometimes even fall short in accurately conveying this relation for common items. Thus, attempts like ReVersion [11] have been made to encapsulate a general object-to-object relation into a special text token, including the relation of "painted on". This is closely related to our purpose. However, upon a detailed examination of ReVersion, we find the "painted on" relation it learns is relatively weak and observe several limitations of this approach: (1) The relation of "painted on" cannot be effectively learned and generalized with too few and similar exemplars as in [11]. (2) The difficulties of learning the relation of "painted on" different objects vary a lot. Therefore, this method might be less effective for logos, since logos have complex details and models lack prior knowledge of logos including the knowledge of where the logos should appear, and how would they interact with other objects in the real world. Hence, we propose the following strategies to tackle these problems and make the learned relation stronger.

Relation data collection. We first collect a diverse dataset of the relation "object1 painted on object2". "Object2s" are subjects that we intend to paint the logo on, such as shirts, hats, and mugs. Instead of real logos, we choose "object1s" to be commonplace objects that the diffusion model possesses more prior knowledge of, like dogs, apples, or stars. We want the relation can be learned more easily and precisely, leveraging the model's encapsulated understandings of how the "object1s" are interacting with these "object2s". The relation dataset consists of 20 different "object2s" and each "object2" has 3 to 4 training images of different poses to ensure diversity. With this diverse dataset, we want the relation of "painted on object2s" to generalize for logos.

Actor-critic sampling. Different objects have different shapes, textures, and rarities. It is natural that painting something onto different objects has varying levels of difficulty. For example, painting patterns on a shirt or a mug is more common and easier than painting patterns on an egg or a slice of toast. Therefore, we need some mechanism to ensure that we can paint the logo more equally onto various "object2s". To this end, we propose to utilize a pre-trained CLIP model [20] as a critic to evaluate the diffusion model's effectiveness in applying common patterns onto different objects, throughout the relation learning process. With some ad-hoc "object1s", we denote the prompt "object1 painted on object2" as c_{obj2} and the prompt using special text token <painted> "object1 <painted> on object2" as $c_{obj2<painted>}$. Then we denote the CLIP score of generating "object1" onto "object2" as s_{obj2} in the following Eq. (2):

$$s_{obj2} = CLIP(LDM(c_{obj2<painted>}), c_{obj2}). \tag{2}$$

Then the Actor-Critic Relation Pre-training algorithm can be summarized in Algorithm 1, which adjusts the sampling probabilities of training images of "object2s" every f iterations. It uses the CLIP critic to rate whether the model has understood how to paint on each "object2". More frequent sampling will be applied on "object2s" that are not mastered, and vice versa. Besides optimizing a single text token, we also fine-tune the text encoder to strengthen the relation.

Algorithm 1 Actor-Critic Relation Pre-training

Require: Diffusion model LDM; Relation text token <painted>, CLIP critic model $CLIP$; constant λ; number of "object2s" N; probability recalibration frequency f; Number of training iterations A.

Ensure: Pre-trained Diffusion Model.

for $a = 1$ to A **do**
  Fine-tune LDM and <painted> on relation dataset with sampling probability $p(obj2)$ for each "object2";
  if $a \mod f = 0$ **then**
    calculate s_{obj2} for each $obj2$ by Eq. (2);
    $\bar{S} = \frac{1}{N}\sum_{obj_2} s_{obj2}$;
    for each $obj2$ **do**
      $w(obj2) = \lambda^{\bar{S}-s_{obj2}}$;
      $p(obj2) = \frac{w(obj2)}{\sum_{obj_2} w(obj2)}$;
    end for
  end if
end for

3.3 Decoupled Identity Learning

The Actor-Critic Relation Pre-training can encapsulate a stronger and more balanced "painted on" relation into diffusion models. Then our next task is to implant logo identities into the output domain of diffusion models. This task is rather challenging due to the model's lack of prior knowledge of diverse and complex logos. Without enough prior knowledge, it is difficult for the model to know which parts of the training images contain the logo to be learned. This easily results in unwanted concepts getting learned and intricate details of the logo not getting captured. We substantiate this issue by showing that usual relevant class names like "logo", "symbol", and "text" elicit no significant attention on diverse logo patterns in training images. In Fig. 3, we provide a visualization of attention maps on common items and logos, using Null-text Inversion [16]. From Fig. 3(b) we can see the attention maps of common items' class name tokens are very accurate. While from Fig. 3(a), we can see that the word "logo" and its synonyms cannot precisely attend the logo regions at all. This challenge stems from the inherent complexity of logos, characterized by their diverse compositions, appearances, patterns, layouts, and textual elements. Consequently, the diffusion model lacks adequate and high-quality shared prior knowledge of logos, which is essential for effective logo recognition. This can make the learning process ambiguous and prone to overfitting to unwanted parts of training images. Thus the fine-grained details of logos cannot be preserved. Hence, to alleviate this issue, we decouple logo identity learning into two parts and propose specific training data generation methods correspondingly.

Logo token binding. Since the diffusion model cannot recognize complex logos, the first and foremost task is to make it able to identify the logo to be learned in images. Due to the model's inability to localize logos in training images, fine-tuning all weights like Dreambooth [25] would cause spurious correlations [12] or

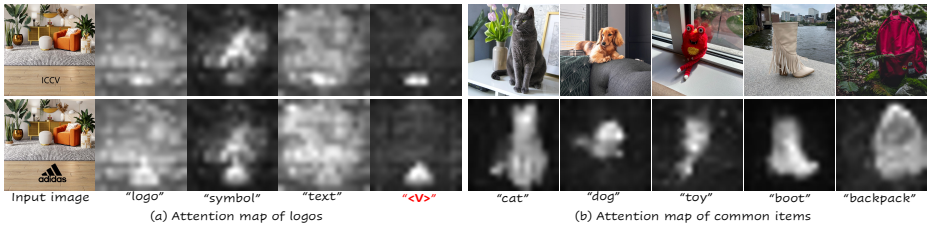

Fig. 3: Visualizations of attention maps for: (a) Common synonyms of the word "logo" and our special token optimized on the logo token binding set. (b) Class name tokens of commonplace items in Dreambooth's [25] dataset. Attention maps are computed by averaging attention activation across time steps and layers.

irrelevant concepts got learned. Therefore, we propose to constrain the learning scope and only optimize a special token <V> using Textual Inversion [8]. Textual Inversion only allows the change of one single text token and thus does not learn much irrelevant information from training images. We want to bind the target logo to <V> so that the logo can be accurately localized in training images. Textual Inversion generally collects training images that present the target concept in diverse settings, including different backgrounds and poses [8]. However, we observe in the case of diverse and complex logos, this setting easily leads to confusion about what concept should be learned. Therefore, we further propose to construct a distinct and less complex logo token binding set by pasting the logo onto random locations of solid color backgrounds contrasting to the color of the logo. We then use Textual Inversion to optimize a text token <V> using our constructed logo token binding set. Although not able to reconstruct the logo, we find the optimized token <V> can elicit significant attention on the logo region of training images, which facilitates further fine-grained logo identity extraction, as shown in Fig. 3(a).

Logo identity learning. With the optimized special token <V> accurately recognizing the target logo in images, our next task is to learn the logo identity accurately. Since the text token is tightly bound with the target logo, we can build a more complicated logo identity learning set. Specifically, we paste the logo onto random locations of various natural scenes with contrasting colors of the logo. Then, with the text token <V> in the prompt, we finetune the weight of the U-Net to distill the logo identity into the special text token precisely.

4 Experiments

In this section, we illustrate our experiment setup, including the dataset we construct, evaluation metrics, and baseline customization methods. Then we compare our method LogoSticker with baseline customization methods quantitatively and qualitatively. Moreover, we present our conducted user studies, comparisons with large text-to-image models DALLE 3 [2], and ControlNet [35]. We also compare LogoSticker with text generation models Textdiffuser-2 [4], and

Table 1: Quantitative results and user study. (Left) Quantitative metric comparisons of identity fidelity (DINO, CLIP-I) and prompt fidelity (CLIP-T). (Right) User study on prompt fidelities (User PF) and logo fidelities (User LF).

Method	DINO ↑	CLIP-I ↑	CLIP-T ↑	User PF ↑	User LF ↑
Textual Inversion [8]	0.228	0.648	0.303	2.44	1.46
DreamBooth [25]	0.227	0.739	0.278	3.01	3.21
DreamBooth + ReVersion [11]	0.232	0.721	0.271	3.03	2.79
LogoSticker (Ours)	0.229	0.761	0.289	4.37	4.46

AnyText [33]. Subsequently, we explore some extensions of our method, demonstrating its versatility. Finally, we present the ablation studies.

4.1 Experimental Setup

Dataset. We collect a dataset of 20 unique logos including texts with patterns, English texts, and Chinese texts with intricate calligraphy.[1] We test the generation of each learned logo identity in 20 different contexts. For quantitative analysis, each logo-context combination is synthesized using 5 different seeds to ensure comprehensive evaluation.

Baselines. To evaluate the effectiveness of our proposed LogoSticker, we compare it with 3 state-of-the-art baselines: Dreambooth [25], Textual Inversion [8], and Dreambooth + ReVersion [11]. We use the official implementation for ReVersion [11], and use the codes from diffusers [19] for Dreambooth and Textual Inversion. For baseline methods, we train them on the logo identity learning set, which consists of logos pasted onto random locations of natural images.

Evaluation metrics. Following prior practices [8, 25], we evaluate our LogoSticker and the baselines using two metrics: prompt fidelity and identity fidelity. Prompt fidelity measures the similarities between text descriptions and the generated images. Identity fidelity measures whether the generated images preserve the logo identity. Prompt fidelity, denoted as CLIP-T, is measured using the cosine distance between the CLIP [20] embeddings of text prompts and generated images. When evaluating prompt fidelity, we replace the special token <V> in the prompt with the word "logo". The identity fidelity, denoted as CLIP-I and DINO, is measured by CLIP [20] and DINO [3] scores correspondingly.

4.2 Comparisons

Quantitative results. Tab. 1 gives quantitative comparisons of identity fidelity and prompt fidelity between our LogoSticker and baselines. From Tab. 1, we can see that LogoSticker outperforms all other methods in preserving logo identity. Existing methods generally have difficulties in learning and maintaining the logo identity. Intriguingly, we find that unlike in previous customization tasks [25],

[1] Existing customization methods generally collect dozens of subjects for evaluation. For example, Dreambooth [25] compiles a dataset of 30 subjects while ReVersion [11] assembles a dataset of 10 subjects.

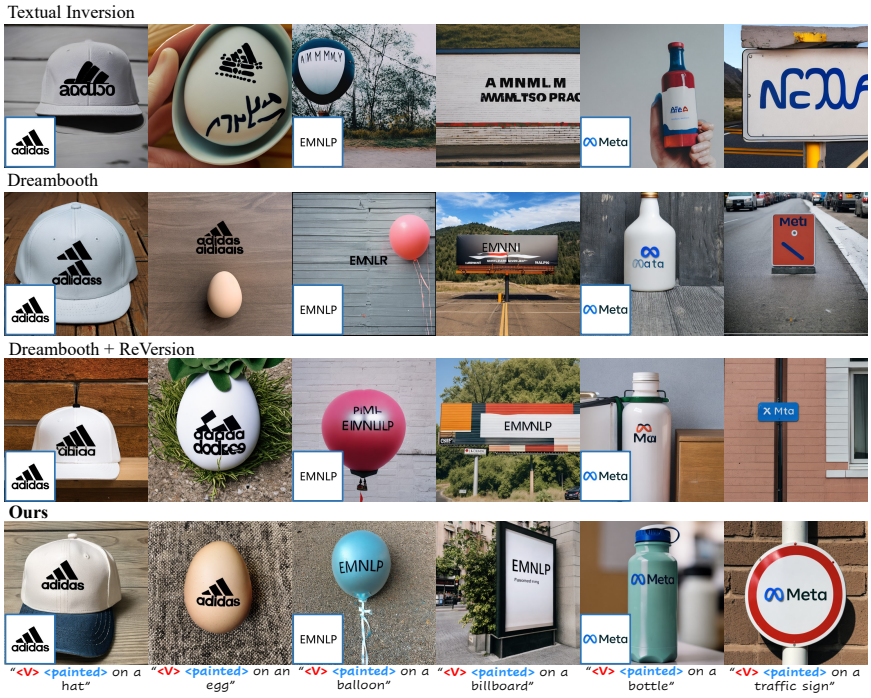

Fig. 4: Qualitative comparisons with baseline customization methods. LogoSticker successfully preserves the logo identity while others struggle. LogoSticker can synthesize the logo coherently on various objects. The logo identity is maintained even on curved objects or under various viewing positions.

where the DINO score is usually close to the CLIP-I score, there exists a significant discrepancy between the DINO score and CLIP-I score for logos. We hypothesize that it might be because the DINO [3] is not a multi-modal model, and thus it is insensitive to textual elements and patterns such as logos. On the other hand, CLIP [20], as a multi-modal model, can recognize texts and patterns quite well. CLIP has also been found to possess some OCR abilities and thus be susceptible to Typographic Attacks [9], which aligns with our guess. We also conduct a user study to further evaluate all methods. The results are presented in Tab. 1, from which we can see that users generally prefer our LogoSticker over others. This further verifies our method's superiority.

Qualitative results. We present the qualitative comparison of 6 logo-context pairs in Fig. 4. From the figure, we can see that Textual Inversion [8] cannot preserve the logo identity. Dreambooth [25] and Dreambooth + ReVersion [11] can preserve the logo identity better, but the logo identity is corrupted when it is generated onto other objects. Instead, our LogoSticker can preserve the logo identity excellently. Also, our generated logos are able to maintain their identities even on curved objects, inclined planes, or under different view angles.

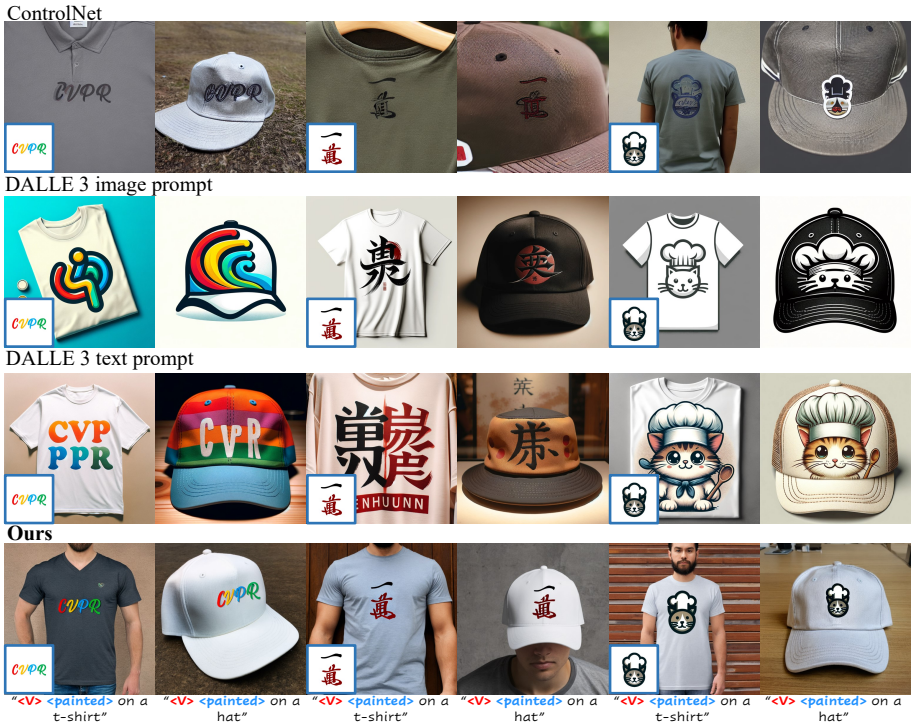

Fig. 5: Qualitative comparisons with large text-to-image models, including ControlNet [35] and DALLE 3 [2] using both detailed text and image prompts.

Comparisons with large text-to-image models. We further compare our LogoSticker with ControlNet [35] and GPT 4's [18] DALLE 3 [2] using both detailed text prompts and uploaded image exemplars to describe target logos. From Fig. 5, we can see that ControlNet can retain the overall shapes of the logos to some level. However, both the details and colors of the logos are corrupted. Also, since ControlNet uses ad-hoc logo positions, the coherence of synthesized images is usually compromised. For DALLE 3 with an image exemplar to describe the logo, the synthesized images can only preserve the high-level ideas of the uploaded logos, while losing all details. For DALLE 3 with detailed text prompts, we can see that it is also not able to preserve the intricate details of the logos using only textual descriptions. Moreover, we can see that DALLE 3 does not have the ability to synthesize legible Chinese texts, and the detailed specification of texts' colors also seems unlikely. In contrast, although built on Stable Diffusion [23], LogoSticker enables coherent generations of logos while accurately maintaining their identities.

Comparisons with text generation models. We also compare LogoSticker with state-of-the-art text generation models Textdiffuser-2 [4] and AnyText [33] using detailed text prompts describing the target text. As demonstrated in Fig.

Fig. 6: Qualitative comparisons with text generation models, including Textdiffuser-2 [4] and AnyText [33].

6, we can see both Textdiffuser-2 and AnyText fail to maintain the fine-grained details of the texts. Neither of them is able to follow the per-character text color specifications and the overall handwritten style. Also, they cannot generate texts together with other concepts like "Darth Vader" or "Michael Jackson". Textdiffuser-2 [4] even has difficulties generating texts onto objects like hats or mugs. In contrast, LogoSticker is free from these issues. It can accurately retain the per-character text color specifications and the overall writing style of the text. The texts can also be painted onto objects with other concepts seamlessly.

4.3 More Applications

More challenging contextualizations. We present more visualizations of our LogoSticker's generations of logos in more diverse and challenging contexts. In Fig. 7, we showcase 3 challenging cases to further validate the strength of LogoSticker. Row 1 of Fig. 7 demonstrates that LogoSticker can paint the customized logo together with other logos onto objects coherently without any additional layout guidance. There are no overlaps or identity entanglements of both logos and identities of both logos are well preserved. Row 2 demonstrates that altering the overall scene in the inference prompt can still result in coherent and identity-preserving generations, while row 3 exemplifies that we can modify the

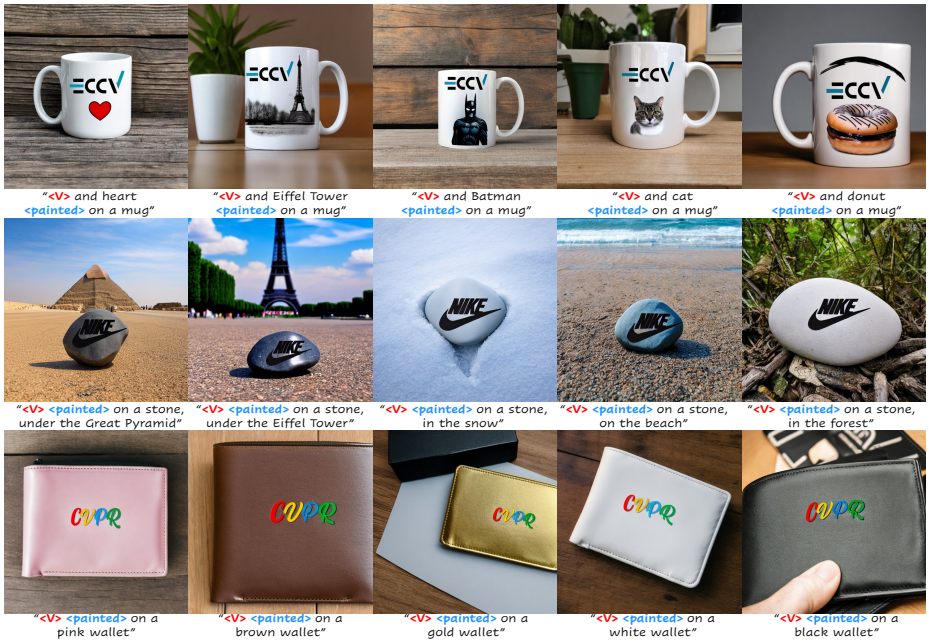

Fig. 7: Application of LogoSticker on more challenging contextualizations.

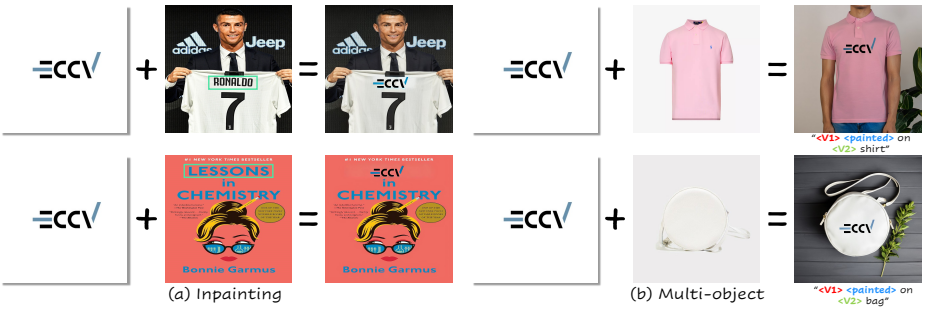

Fig. 8: Application of LogoSticker on: (a) Inpainting. (b) Customizing both the logo and the context.

properties of objects being painted on without hurting the logo's fidelity. These observations demonstrate that coherent and visually pleasing synthesis can be obtained with identity-preserved logos using LogoSticker.

Inpainting. LogoSticker adapts well with inpainting [23]. We can directly plug it into existing inpainting pipelines [23]. In this way, LogoSticker is able to inpaint logos on user-provided images. From Fig. 8(a), we can see that the logo identity is well-preserved while the background identity is also maintained.

Multi-concept customization. We conduct a preliminary experiment on LogoSticker's compatibility with multi-concept customization. We simply integrate additional training images of the context into the training set during the logo identity learning phase. From Fig. 8(b), we can see that the identities of the polo shirt and round bag are well-preserved and the logo is painted on them seamlessly. This indicates the possibility of combining LogoSticker with advanced multi-concept customization methods for more complex customization tasks.

4.4 Ablation Studies

Next, we conduct ablation studies to examine the effect of each component of our LogoSticker qualitatively and quantitatively. We first ablate the effect of the CLIP critic during our relation pre-training. As illustrated in row 1 of Fig. 9(a) and Fig. 9(b), painting logos onto some objects can be difficult and not adequately learned without the CLIP critic. Although the model learns the identity of the logo, it does not know how to paint the logo onto certain objects. Quantitatively, it is reflected as the decreases in prompt and logo fidelities. Qualitatively, when prompted to generate a logo on certain objects, the model might generate the logo somewhere else or synthesize a logo with an incomplete identity.

We then ablate the effect of the logo token binding set. From row 2 of Fig. 9(a) and Fig. 9(b), we can see that, without the logo token binding set, the model cannot precisely recognize the target logo and thus cannot extract the logo identity accurately. The model would be confused and tend to generate multiple or incomplete representations of the logo.

(a) Qualitative ablation studies

(b) Quantitative ablation studies

Method	CLIP-I ↑	CLIP-T ↑
w/o CLIP critic	0.717	0.266
w/o Logo token binding set	0.742	0.275
Ours	0.755	0.275

Fig. 9: Ablation study. (a) Qualitative (b) Quantitative ablation studies on the use of: (1) the CLIP critic for relation pre-training; (2) the logo token binding set for logo token binding.

5 Conclusion

In this work, we conduct a pioneering exploration and introduce a novel task of logo insertion, aiming to insert diverse and complex logos accurately into diffusion models and enable identity-preserving and coherent generation of these logos in various contexts. We propose an effective two-phase pipeline LogoSticker for tackling this task, consisting of the actor-critic relation pre-training algorithm followed by the decoupled identity learning algorithm. LogoSticker exhibits superior fidelities and coherence when generating diverse logos in various contexts, showing that the customization method can effectively deal with challenging logos, which encompass multilingual textual elements and intricate patterns. We hope our approach can benefit real-world applications like advertising.

Acknowledgement. This work was supported in part by the Research Grants Council under the Areas of Excellence scheme grant AoE/E-601/22-R, the National Natural Science Foundation of China (No. 62201484), HKU Startup Fund, and HKU Seed Fund for Basic Research.

References

1. Avrahami, O., Aberman, K., Fried, O., Cohen-Or, D., Lischinski, D.: Break-a-scene: Extracting multiple concepts from a single image. arXiv:2305.16311 (2023) 1, 4
2. Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., Manassra, W., Dhariwal, P., Chu, C., Jiao, Y., Ramesh, A.: Improving image generation with better captions. https://openai.com/dall-e-3 (2023) 1, 3, 4, 5, 8, 11
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021) 9, 10
4. Chen, J., Huang, Y., Lv, T., Cui, L., Chen, Q., Wei, F.: Textdiffuser-2: Unleashing the power of language models for text rendering. arXiv:2311.16465 (2023) 3, 4, 8, 11, 12
5. Chen, J., Huang, Y., Lv, T., Cui, L., Chen, Q., Wei, F.: Textdiffuser: Diffusion models as text painters. In: NeurIPS (2023) 1, 4
6. Chen, X., Huang, L., Liu, Y., Shen, Y., Zhao, D., Zhao, H.: Anydoor: Zero-shot object-level image customization. arXiv:2307.09481 (2023) 1, 4
7. Gafni, O., Polyak, A., Ashual, O., Sheynin, S., Parikh, D., Taigman, Y.: Make-a-scene: Scene-based text-to-image generation with human priors. arXiv:2203.13131 (2022) 1
8. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. In: ICLR (2023) 1, 2, 4, 8, 9, 10
9. Goh, G., Cammarata, N., Voss, C., Carter, S., Petrov, M., Schubert, L., Radford, A., Olah, C.: Multimodal neurons in artificial neural networks. Distill (2021) 10
10. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020) 4, 5
11. Huang, Z., Wu, T., Jiang, Y., Chan, K.C.K., Liu, Z.: Reversion: Diffusion-based relation inversion from images. arXiv:2303.13495 (2023) 2, 4, 6, 9, 10
12. Joshi, S., Yang, Y., Xue, Y., Yang, W., Mirzasoleiman, B.: Towards mitigating spurious correlations in the wild: A benchmark and a more realistic dataset. arXiv:2306.11957 (2023) 7
13. Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up gans for text-to-image synthesis. arXiv:2303.05511 (2023) 1
14. Kumari, N., Zhang, B., Zhang, R., Shechtman, E., Zhu, J.Y.: Multi-concept customization of text-to-image diffusion. In: CVPR (2023) 1, 4
15. Liu, Z., Zhang, Y., Shen, Y., Zheng, K., Zhu, K., Feng, R., Liu, Y., Zhao, D., Zhou, J., Cao, Y.: Cones 2: Customizable image synthesis with multiple subjects. arXiv:2305.19327 (2023) 4
16. Mokady, R., Hertz, A., Aberman, K., Pritch, Y., Cohen-Or, D.: Null-text inversion for editing real images using guided diffusion models. arXiv:2211.09794 (2022) 7
17. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv:2112.10741 (2022) 1

18. OpenAI: GPT-4 technical report. arXiv:2303.08774 (2023) 11
19. von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Wolf, T.: Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers (2022) 9
20. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021) 3, 4, 6, 9, 10
21. 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. JMLR **21**(140), 1–67 (2020) 4
22. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv:2204.06125 (2022) 1, 4
23. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. arXiv:2112.10752 (2021) 1, 4, 5, 11, 13
24. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. arXiv:1505.04597 (2015) 5
25. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: CVPR (2023) 1, 2, 4, 7, 8, 9, 10
26. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding. In: NeurIPS (2022) 1, 4
27. Sarukkai, V., Li, L., Ma, A., Ré, C., Fatahalian, K.: Collage diffusion. arXiv:2303.00262 (2023) 4
28. Sauer, A., Karras, T., Laine, S., Geiger, A., Aila, T.: Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. arXiv:2301.09515 (2023) 1
29. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv:2111.02114 (2021) 5
30. Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. arXiv:1503.03585 (2015) 4
31. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: ICLR (2021) 4, 5
32. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: ICLR (2021) 4, 5
33. Tuo, Y., Xiang, W., He, J.Y., Geng, Y., Xie, X.: Anytext: Multilingual visual text generation and editing. arXiv:2311.03054 (2024) 3, 4, 9, 11, 12
34. Yang, Y., Gui, D., Yuan, Y., Liang, W., Ding, H., Hu, H., Chen, K.: Glyphcontrol: Glyph conditional controllable visual text generation. In: NeurIPS (2023) 4
35. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. arXiv:2302.05543 (2023) 4, 8, 11