Action Imitation in Common Action Space for Customized Action Image Synthesis

Wang Lin[1,*], **Jingyuan Chen**[1,†], **Jiaxin Shi**[2,*], **Zirun Guo**[1], **Yichen Zhu**[1], **Zehan Wang**[1], **Tao Jin**[1], **Zhou Zhao**[1], **Fei Wu**[1], **Shuicheng YAN**[3], **Hanwang Zhang**[4]
Zhejiang University[1], Xmax.AI[2], Skywork AI Singapore[3], Nanyang Technological University[4]
{linwanglw, jingyuanchen, gzr, yc_zhu, wangzehan01,
jint_zju, zhaozhou, wufei}@zju.edu.cn
shijx12@gmail.com, shuicheng.yan@kunlun-inc.com,
hanwangzhang@ntu.edu.sg

Abstract

We propose a novel method, **TwinAct**, to tackle the challenge of decoupling actions and actors in order to customize the text-guided diffusion models (TGDMs) for few-shot action image generation. TwinAct addresses the limitations of existing methods that struggle to decouple actions from other semantics (*e.g.*, the actor's appearance) due to the lack of an effective inductive bias with few exemplar images. Our approach introduces a common action space, which is a textual embedding space focused solely on actions, enabling precise customization without actor-related details. Specifically, TwinAct involves three key steps: 1) Building common action space based on a set of representative action phrases; 2) Imitating the customized action within the action space; and 3) Generating highly adaptable customized action images in diverse contexts with action similarity loss. To comprehensively evaluate TwinAct, we construct a novel benchmark, which provides sample images with various forms of actions. Extensive experiments demonstrate TwinAct's superiority in generating accurate, context-independent customized actions while maintaining the identity consistency of different subjects, including animals, humans, and even customized actors. Project page: https://twinact-official.github.io/TwinAct/.

1 Introduction

We are interested in customizing Text-Guided Diffusion Models (TGDMs) (*e.g.*, Stable Diffusion [21]) to generate customized actions specified by a few user-provided images. For example, given a few images of *Black Widow*'s signature three-point landing action, we can represent this action as a unique token V^* and generate imaginary creation by the prompt like *"Leonardo V^*"* (the third row in Figure 1). One might question why we do not condition TGDMs on precise textual descriptions [21] or sketch images [31] to generate the desired action. The reason is that actions are often unutterable, and even with finely detailed descriptions (SD column in Figure 1) or skeleton images (ControlNet column in Figure 1), TGDMs cannot accurately follow these instructions.

Further, can we use existing few-shot customized TGDMs [4, 7, 8, 10, 14, 22, 27] to customize actions? Unfortunately, the answer is still no. This is because these methods fine-tune the textual embeddings of V^* in prompts like "*Black Widow* V^*" by reconstructing the corresponding image. But, except for actions, there are many other semantics in the exemplar images that are coupled in V^*, like the characteristics of the specific actors. Thus, the few-shot action images in those methods

[*] Equal Contribution.
[†] Corresponding Author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-0390

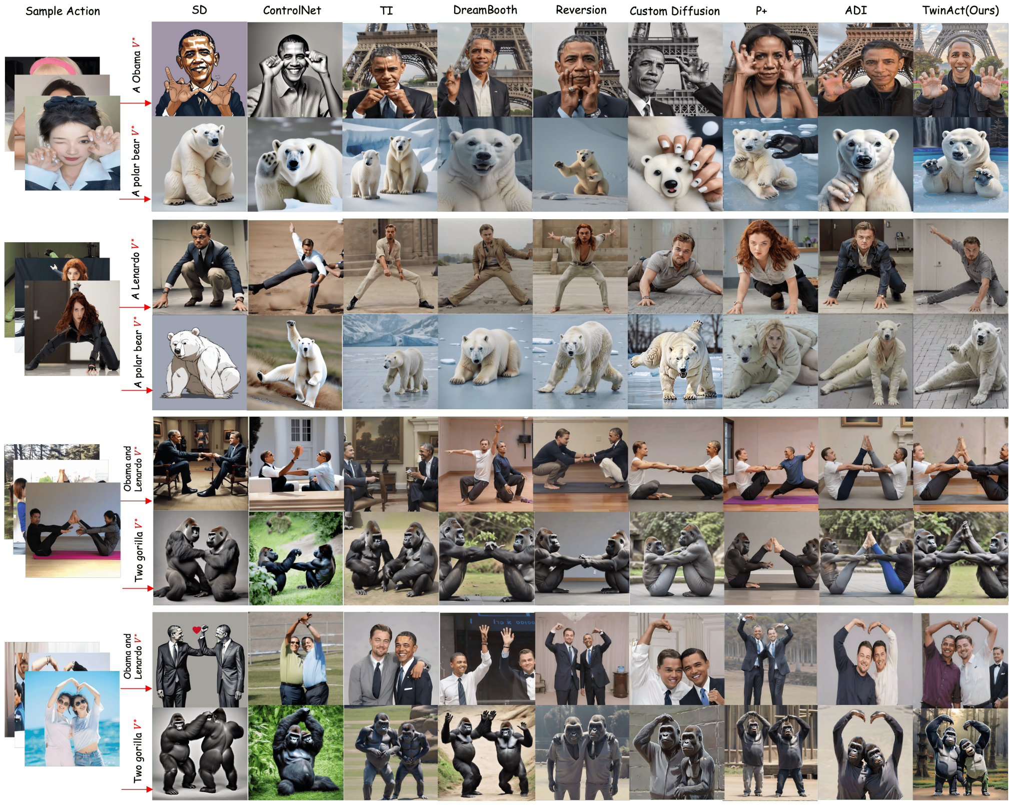


Figure 1: **Qualitative comparisons of TwinAct with other methods.** TwinAct preserves the identity consistency of actors while allowing customized actions to be accurately generalized across different actors by effectively decoupling actions and actors.

lack an effective inductive bias to achieve decoupled action tokens V^* from actors, which results in action-irrelevant information like *Black Widow*'s red curly hair being confounded in the generated images as shown in Figure 1. Although some methods mitigate the effects of missing inductive bias through contrastive learning on textual embeddings or visual images, they still struggle to exclude all action-irrelevant semantics. For example, [8] uses prepositions as positive samples and randomly selects negative samples from a limited set of POS words (*i.e.*, nouns, adjectives). However, this approach can not guarantee the exclusion of all action-irrelevant information from exemplar images for V^*, resulting in the inclusion of irrelevant semantics like *manicure* and *red curl hair* (the first and third rows in Figure 1) confounding the customized tokens. On the other hand, [7] and [14] utilize contrastive training data involving the same subject with and without specific customized concepts (*e.g,* melted and closed eyes, etc.) to decouple the subject and customized concepts. However, the creation of extensive contextual training data is costly, particularly when dealing with customized actions that involve multiple subjects.

By observing the real-world process of a director instructing an actor to perform a specific action, such as the signature action of the character *Black Widow* which involves a set of action phrases including "left hand on the ground", "right hand raised behind the back", "left leg on the ground", and "right leg straight out", we find that these phrases concentrate solely on actions without providing any specific details about the actor involved. Moreover, these phrases can be modified or combined to create new actions, as illustrated in Figure 2. Inspired by these observations, we propose a method called **TwinAct**, which aims to decouple the action from the actor by constructing an effective *Common Action Space*. This space is exclusively for actions, eliminating any confusion with actor-specific characteristics and enabling the generation of diverse and creative actions that can be applied across various actors and scenarios.

Our approach involves three steps. **1)** We collect 832 action phrases that cover a range of movements from finger gestures to full body poses, and extract the corresponding textual embeddings using the tokenizer of CLIP [18]. These textual embeddings are used to build a common action space (Section 3.2) through Principal Component Analysis (PCA [9]); **2)** We employ a Multilayer Perceptron (MLP) to modulate the PCA coefficients to imitate the customized actions within the action space (Section 3.3); and **3)** We introduce an action similarity loss to provide semantic similarity between the exemplar images and generated images, resulting in synthesizing highly adaptable customized action images in diverse contexts (Section 3.3). Extensive experiments demonstrate the superiority of our method over the existing method. Finally, the main contributions of this paper are summarized as:

- We propose **TwinAct** to generate customized action images, which can synthesize novel renditions of user-specific action in different contexts including animals, humans, and even customized actors.
- We introduce a common action space that focuses solely on the action itself, without any actor-specific details. By combining action bases in this space for customized action imitation, we are able to effectively decouple action from actor, allowing for greater flexibility in customization.
- Through extensive experiments, we demonstrate that TwinAct outperforms previous methods in maintaining the fidelity of customized actions while also preserving the consistency of actor identities.

2 Related Work

2.1 Image Generation and Customization

The text-guided diffusion models (TGDMs) [15, 19, 20, 21, 23] have emerged as a powerful paradigm in the domain of image generation, enabling the creation of image variations that are aligned with specific textual descriptions. Recently, the customized TGDMs [4, 5, 8, 10, 13, 11, 22] aim to encoder the customized concept within the textual embedding, often represented by a unique token and decode the token into pixels by the cross-attention in the U-net decoder [24]. Textual Inversion [4] optimizes textual embedding and synthesizes personalized images by integrating the concept token with the target prompt. DreamBooth [22] extends this concept by proposing a framework that optimizes all parameters of the denoising U-Net architecture, based on a specific token and the class category of the subject. Several other works [5, 10] have focused on optimizing subsets of weights or introducing additional adapters to achieve more efficient optimization and better conditioning of the generated images. For instance, Custom Diffusion [10] fine-tunes only the cross-attention layers within the U-Net, while P+ [27] expands the textual-conditioning space with per-layer tokens to allow for greater disentanglement and control over the generation process. Despite these methods achieving commendable results in customized actors, they struggle with generating customized action images, as shown in Figure 1, due to the lack of an effective inductive bias with few exemplar images.

2.2 Customized Action Image generation

In contrast to the rapid progress in customized actors, customized action has received less attention in the community. Since the complex details such as specific angles, body positions, and motion trajectories, describing a specific action is more challenging than describing a specific actor. Even with finely detailed descriptions (see Appendix A.1) TGDMs still cannot accurately follow these textual descriptions. A straightforward approach to generate a specific action image is ControlNet [31] which is conditional on a given skeleton image to generate images. However, they only provide a rough approximation of action consistency (see Appendix A.3). Particularly, when it comes to capturing fine details such as fingers, skeleton images may lack the necessary detail to accurately reproduce these actions, leading to noticeable visual defects in the generated images. In addition, when conditioned on detailed images (see Appendix A.3), they suffer from limited diversity and flexibility. Recent advancements for customized TGDMs preliminary explore action-based customization. Reversion [8] proposes relation-steering contrastive learning that aims to guide the relation prompt towards relation-dense regions within the text embedding space, thereby disentangling the learned relation from other concepts like appearances. Lego [14] and ADI [7] construct contexts involving the same subject with and without a specific concept, to decouple the subject and specific concepts. However, these methods by employing contrastive learning on textual embeddings or visual images can not effectively decouple actions from actors. This is because they cannot exclude all irrelevant concepts with a limited number of negative samples.

https://doi.org/10.52202/079017-0390

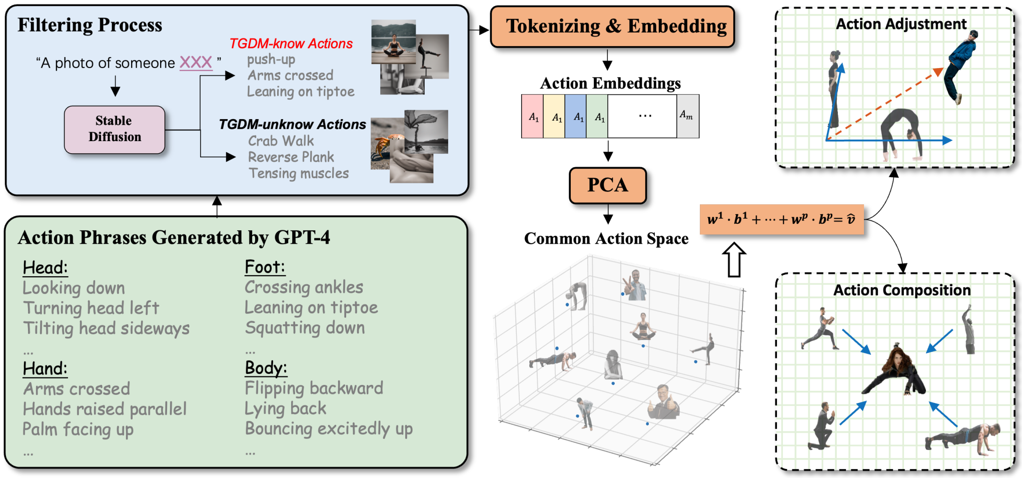


Figure 2: **The construction of Common Action Space.**

3 Methods

Imagine the following scenario: When the character *Black Widow* performs her signature three-point landing action, the director may give the following instruction "*Bend knees and lower body down into a squatting position; Place right hand forward, and right arm should be fully extended*". We can see that when people are asked to perform a specific action, it is often accompanied by instructions that involve the composition of various body movements and the adjustment of angles and amplitudes as shown in the right part of Figure 2. Inspired by the way in which humans perform specific actions, we propose **TwinAct** which applies similar principles to action customization. Specifically, we first collect a set of basic actions and encode them using a pre-trained text encoder to build a common action space represented by a set of action bases. Subsequently, we imitate the customized action by decomposing it into a combination of action bases with corresponding weights. Finally, we utilize both the reconstruction loss and action similarity loss to fine-tune the customized token embedding in the text encoder and the LoRA [6] layer in the text-to-image decoder to generate high-fidelity images of the customized action.

3.1 Preliminaries

Our study is based on the text-guided diffusion models (TGDMs) which consist of a textual encoder $\mathcal{E}$ and a text-to-image decoder $\mathcal{D}$. Given a few sample action images $\mathbf{x}$ and a textual prompt c with a customized token V^*, such as "a photo of V^*", we aim to fine-tune the embedding of customized token V^* in $\mathcal{E}$ and the LoRA layer in $\mathcal{D}$ to realize action customization. The textual encoder $\mathcal{E}$ first tokenizes the prompt c into a set of token embeddings $\mathbf{h}$, which are then used to generate a textual condition $\tau_\theta(c)$ with the text transformer in $\mathcal{E}$. Subsequently, the textual condition $\tau_\theta(c)$ is utilized by the conditional denoising diffusion model $\epsilon_\theta(\cdot)$ in $\mathcal{D}$ to generate images. The commonly used training objective is:

$$\mathcal{L}_{rec} = \mathbb{E}_{\epsilon,t,\mathbf{x}_0,c}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t, \tau_\theta(c))\|^2\right] \tag{1}$$

where t is the timestep, $\mathbf{x}_0$ is the original image and $\mathbf{x}_t$ is a noisy image constructed by adding noise $\epsilon \sim \mathcal{N}(0,1)$ to $\mathbf{x}_0$. After training, any textual condition $\tau_\theta(c)$ with V^* can generate customized action in a new context defined by prompt c.

3.2 Step-1: Building Common Action Space

The initial step involves identifying the set of action bases that defines the common action space within the text-to-image diffusion model.

First, we task GPT-4 [1] with generating a range of common action phrases based on different body parts, including head, fingers, hands, and full body (see Appendix B.2). Then, we carefully sift through the generated action phrases to eliminate duplicates. Subsequently, we build a manual filter based on the pre-trained text-to-image diffusion model by constructing prompts and synthetic images

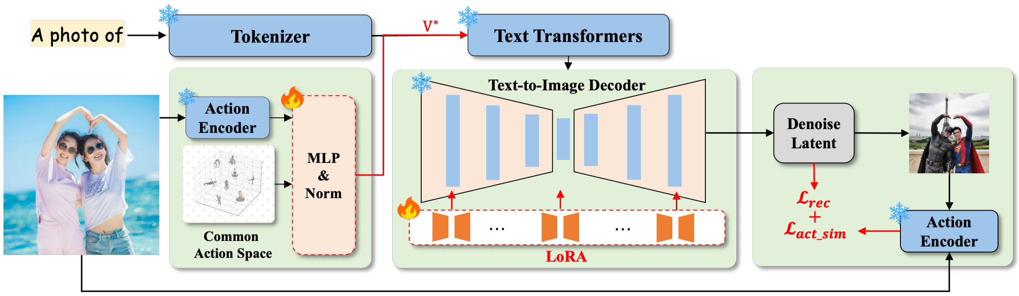


Figure 3: **The overview of the proposed TwinAct.** We optimize the coefficients of the action bases to avoid encoding the action-irrelevant features. After training, we combine the learned coefficients and shared action base to generate images with the customized action.

of each action phrase to remove those action phrases that are unknown to the text-to-image diffusion model, *e.g.,* "Reverse Plank" as shown in Figure 2.

After the filtering process, we identify a total of $m = 832$ action phrases. Each phrase a_i, where $i \in \{1, ..., m\}$, is tokenized and encoded into an action embedding group $A_i = [a_1^i, ..., a_k^i]$. It is important to note that the length k of each action embedding group A_i varies, as each action may consist of a different number of words. Through empirical observation, it was found that action phrases with three words or fewer are the most common, prompting the designation of each A_i as containing three embeddings (*i.e.*, $k = 3$ for all m actions). For simplicity, we denote the three embeddings of all A_i as Q_1, Q_2, and Q_3 respectively, where $Q_k = [a_k^1, ..., a_k^m]$.

Inspiring by prior research [2] employing PCA to transform high-dimensional facial data into a lower-dimensional space, we calculate the PCA transformation matrix B_k for each set VQ_k using the function PCA(Q_k, p). This transformation plays a crucial role in reducing redundant information within the data and enhancing the efficiency of model optimization. Specifically, PCA(Q_k, p) reduce dimensionality of $Q_k \in \mathbb{R}^{m\times d}$ and generate p principal components as $B_k = [b_k^1, \dots, b_k^p] \in \mathbb{R}^{p\times p}$, which captures the essential features of data in a more compact form. Our experimental findings indicate that optimal results are achieved when $p = 512$ (see Section 4.3). Finally, we obtain a set of action bases as $[B_1, B_2, B_3]$, which define an action space that does not contain any action-irrelevant information thus providing a strong inductive bias to achieve decoupled action token embeddings.

3.3 Step-2: Imitating Action via Action Bases

This step involves searching for the optimal parameters to combine the action bases for imitating customized actions.

Our experiments (see Figure 1) have shown that existing methods, which aim to identify optimal personalized token embeddings within the vast text space of pre-trained text encoders, struggle with the inclusion of non-action information. And even with contrastive learning, they are still unable to exclude all irrelevant information. Fortunately, action bases offer an alternative approach for optimizing the customized token embedding. By combining action bases $[B_1, B_2, B_3]$ with corresponding coefficients $[W_1, W_2, W_3]$, where $W_k = [w_k^1, \dots, w_k^p]$, new actions can be imitated as shown in Figure 2. To achieve this, we first utilize a Multi-Layered Perceptron (MLP) to combine the action bases. While backpropagation is a common method for determining optimal coefficients, our experiments indicate its limited effectiveness due to the scarcity of user-provided images (see Section 4.3). Therefore, we use CLIP as an action encoder to extract semantic features as a prior. Specifically, given an action sample image $\mathbf{x}$, we first extract the action features $\mathbf{x}_p$ with action encoder $\psi(\mathbf{x})$, then MLP is used to map $\mathbf{x}_p$ into the modulating coefficients $[W_1, W_2, W_3]$ as:

$$W_k = \mathtt{MLP}(\mathbf{x}_p) \tag{2}$$

Finally, for a customized action, we can combine the action bases to imitate the action as:

$$\hat{V} = [\hat{v}_1, \hat{v}_2, \hat{v}_3], \hat{v}_k = \sum_{j=1}^{p} w_k^j b_k^j \tag{3}$$

https://doi.org/10.52202/079017-0390

where $\hat{V}$ is the embedding of the customized tokens V^*. In this way, $\hat{V}$ is a linear combination of the action bases. Consequently, the vector interpolated in the action space would not contain any action-irrelevant features, allowing us to achieve the objective of decoupling the action and the actor.

3.4 Step-3: Generating Customized Action

The final step involves optimizing the embeddings of the customized token V^*, which is derived from the action base through linear combinations and synthesis customized action image.

Training. Initially, a few user-provided images and corresponding textual prompts containing V^* are used to apply the pixel reconstruction-based loss $\mathcal{L}_{rec}$, similar to previous customized TGDMs (in Eq 1). However, the traditional $\mathcal{L}_{rec}$ in existing TGDMs aims to faithfully reconstruct the sample image appearance, focusing on low-level pixel features. This narrow focus limits the model's ability to recognize and invert high-level action features in the images, as shown in Figure 7(b).

To address this limitation, we introduce an action similarity loss to enhance the model's grasp of high-level action semantics. Beginning with an input image $\mathbf{x}$ and a corresponding prompt c, we randomly initialize a latent variable $\mathbf{x}_T$. This variable is progressively denoised until it reaches a randomly selected timestep t, at which point the denoised image $\mathbf{x}^{prd}$ is predicted directly from $\mathbf{x}_t$. The goal is to evaluate the action similarity between these two images. Specifically, by leveraging action encoder $\psi(\cdot)$, the encoded action features for both the reference action $\mathbf{x}_p$ and the generated action $\mathbf{x}_p^{prd}$ are extracted. We then calculate the cosine similarity between these action features to measure action consistency during the generation process. This similarity is formally defined as:

$$\mathcal{L}_{act\_sim} = \mathbb{E}_{c\sim p(c)}\mathbb{E}_{x\sim p(x|c)}\left[1 - \text{cos\_sim}(\mathbf{x}_p, \mathbf{x}_p^{prd})\right] \tag{4}$$

By integrating the action similarity loss, the model's focus shifts from detailed pixel information to high-level action semantics. The final training loss for TwinAct can be expressed as:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{act\_sim} \tag{5}$$

Testing. After training, the three groups of coefficients $[W_1, W_2, W_3]$ and the LoRA parameters in the text-to-image decoder are saved. Users can generate images depicting customized actions in different contexts by providing prompts such as *"Leonardo V^*"*.

4 Experiment

4.1 Experiment Setup

Dataset. We tackle the challenge of decoupling specific actions from the actors in user-provided images. Since there are no publicly available customized action datasets, we introduce a novel benchmark, consisting of 12 actions involving multiple body parts, such as fingers, arms, legs, and full-body motions. Each action contains approximately 10 sample images. In addition to actions performed by a single actor, we incorporate more complex actions that involve multiple actors.

Metrics. To comprehensively validate the efficacy of TwinAct, we incorporate objective metrics to assess the quality of the generated images. We calculate the consistency between the action in the sample image and the generated image through the CLIP score, which is denoted as "S_{Action}". Additionally, ensuring identity consistency is a crucial aspect of our task, so we evaluate the actor's identity similarity using a pre-trained face recognition encoder [3], which is denoted as "S_{Actor}". In addition, we also conduct user studies including "U_{Action}" and "U_{Actor}", representing user preferences for action fidelity and actor identity consistency in the generated images.

4.2 Comparison with State-of-the-art Methods

Qualitative Comparison. We illustrate the advancements of our method compared to previous approaches in handling various forms of actions, including hand, body, single-actor, and multi-actor actions, as shown in Figure 1. Initially, we evaluate two methods for generating customized action images without fine-tuning, as shown in the column of SD and ControlNet in Figure 1. Images generated from detailed textual prompts (SD) frequently exhibit misalignment with the actions in the sample images. On the other hand, Controlled Generative Models (ControlNet), which depend on

https://doi.org/10.52202/079017-0390

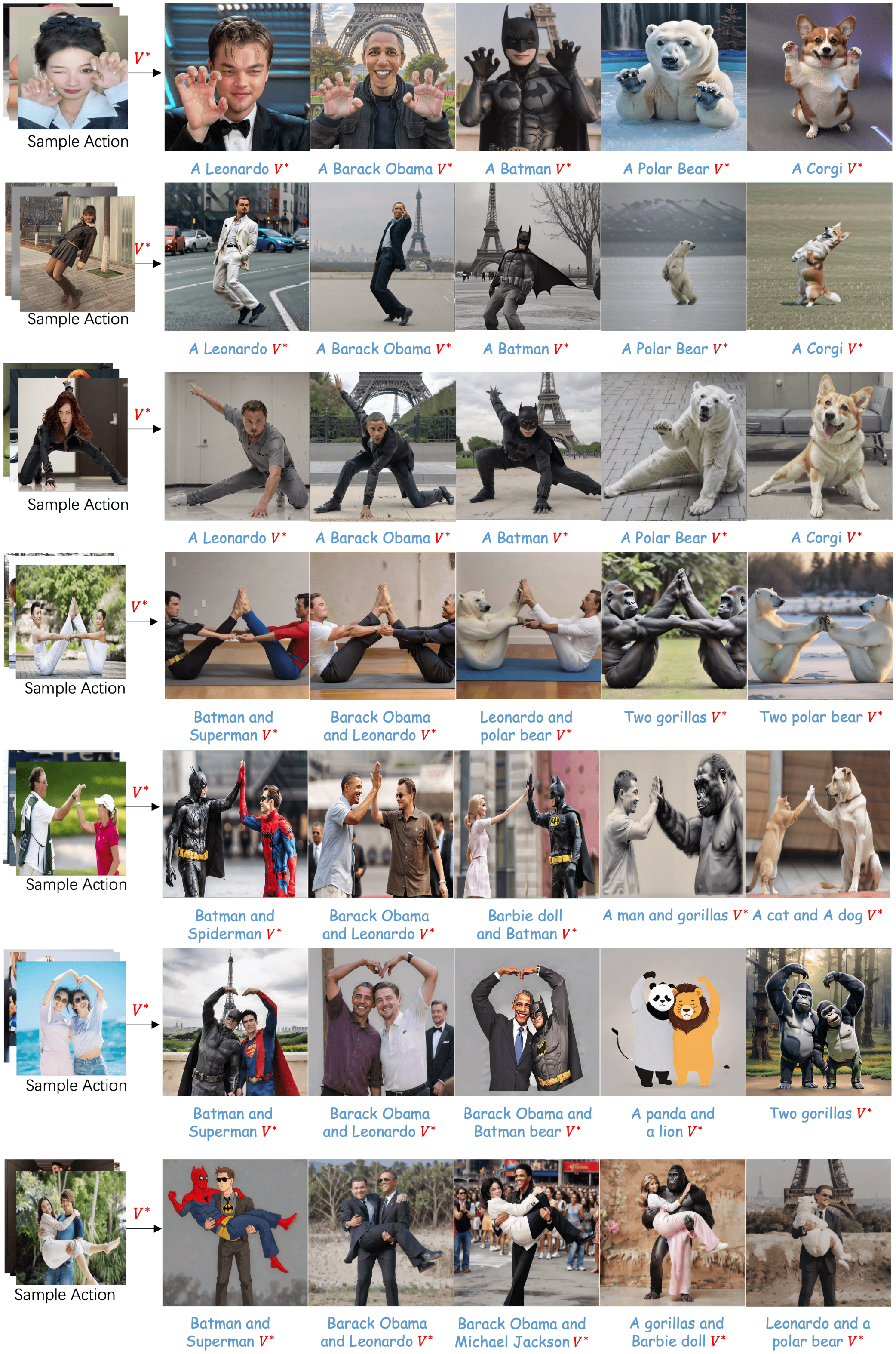


Figure 4: **The results of customized action generation with TwinAct.** TwinAct generates images of different actors performing customized actions such as celebrities and animals, and maintains the consistency of the action and identity of the subject.

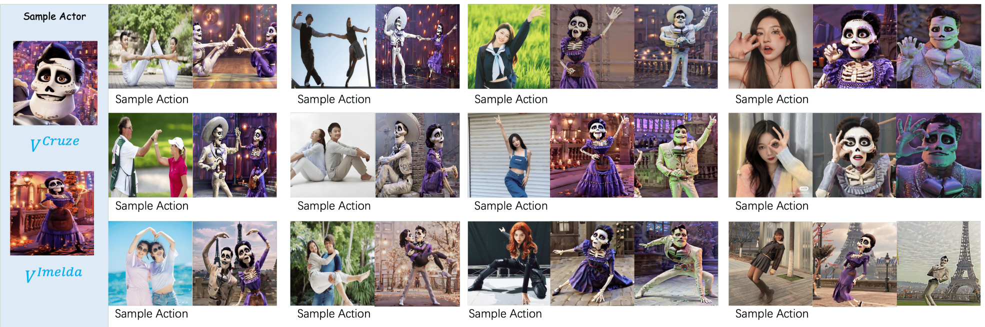


Figure 5: The results of **customized actors** performing **customized actions** generated by TwinAct.

Methods		Text Inversion	DreamBooth	Reversion	Custom Diffusion	P+	ADI	Ours
Objective	S_{Action}	9.12	12.23	18.73.	26.83	33.95	45.32	**69.47**
	S_{Actor}	33.85	38.53	45.58	41.58	46.12	48.76	**73.34**
User Study	U_{Action}	0.22	0.19	0.89	1.16	1.30	2.25	**3.86**
	U_{Actor}	0.53	0.70	1.05	1.31	1.23	2.08	**4.12**

Table 1: **Quantitative comparisons.** We evaluate TwinAct based on both objective metrics and user study. The results indicate that TwinAct surpasses all baseline methods.

skeleton images, tend to only imitate the basic structures of the sample images and lack a profound understanding of the contextual dependencies required for action customization. This deficiency leads to significant body deformations and flaws, particularly when animals are involved in the action. For example, in the fifth and sixth rows of Figure 1, both the human and the gorilla mistakenly raise their hands instead of their feet.

Subsequently, we conduct comparisons with state-of-the-art baselines such as Textual Inversion [4], DreamBooth [22], Reversion [8], Custom Diffusion [10], P+ [27], and ADI [7]. Textual Inversion and DreamBooth tend to capture minimal action-related information, primarily because of their limited tunable parameters. On the other hand, Custom Diffusion, which fine-tunes cross-attention in U-net, and P+, which utilizes per-layer token embeddings, manage to retain more information from the sample images. However, they uncontrollably invert action-irrelevant details in the sample images and bring these details into the generated image, such as the human hand and the manicure in the first and second rows shown in Figure 1. In addition, Reversion and ADI try to exclude action-irrelevant information from the images by contrastive learning but face challenges in comparing all negative samples. Therefore, the images they generate still contain some confounded details, such as the red curly hair of the *Black Widow* in the third row and the yoga pants in the sixth row. In contrast, TwinAct shows the best customized action generation results. Through the acquisition of a customized action token embedding, which is derived from a linear combination of action bases within the common action space, TwinAct accurately capture the action information in the sample images while excluding irrelevant details with good generalization.

Quantitative Comparison. In addition to the visual quality, we also present a numerical comparison between TwinAct and the baselines in Table 1. As can be seen, our approach outperforms in terms of both action and actor similarity, demonstrating its effectiveness in imitating customized actions and filtering out irrelevant information that could potentially confuse the portrayal of the new actor. Furthermore, the performance across four representative types of action customization tasks (*i.e.*, Single-Actor, Two-Actor, With-Human, With-Animal) is illustrated in Figure 6. Our method consistently shows superior performance and the least variability across all types. Notably, the method labeled as P+ experiences the most significant performance drop when involving animals, and ADI exhibits a considerable performance drop when handling actions involving two actors.

Additionally, following the methodology outlined in [30], we conduct a quantitative comparison involving human evaluators to further validate the effectiveness of TwinAct. We invite 100 users to rate each pair of actor-action images generated by TwinAct and baselines on a scale ranging from 1 (worst) to 5 (best), focusing on both action and actor consistency. The collective feedback, as presented in Table 1, clearly indicates a strong preference for TwinAct among the users.

https://doi.org/10.52202/079017-0390

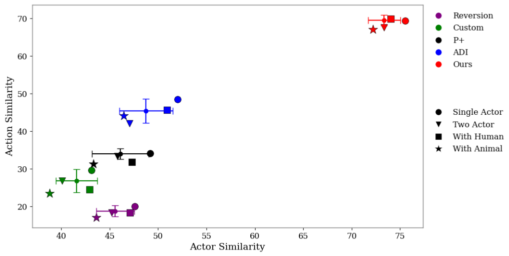
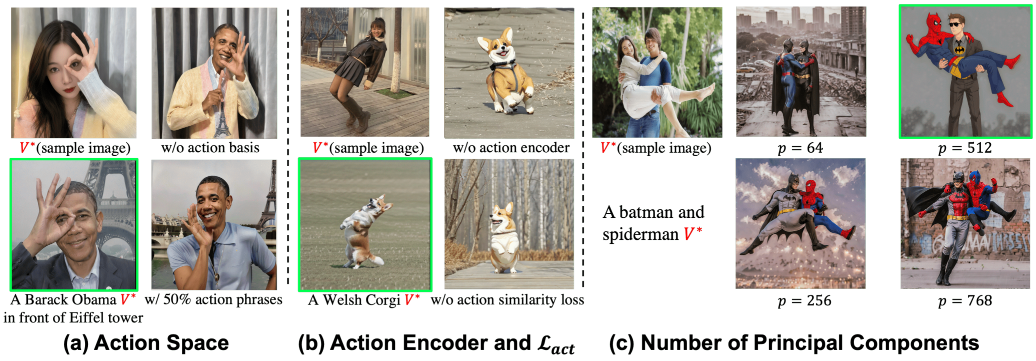


Figure 6: Numerical analysis in terms of the actor and action similarity on four prompt types.

Methods	S_{Actor}	S_{Action}
w/o action space	58.86	52.18
w/ 50% action phrases	68.39	61.86
w/o action encoder	67.27	59.06
w/o $\mathcal{L}_{act}$	70.23	63.58
$p = 64$	72.47	54.27
$p = 256$	73.12	61.66
$p = 768$	71.62	67.63
Ours	**73.34**	**69.47**

Table 2: Ablation studies.



(a) Action Space **(b) Action Encoder and $\mathcal{L}_{act}$** **(c) Number of Principal Components**

Figure 7: **Visualization of ablation studies.** The results of TwinAct are highlighted in green box.

4.3 Ablation Study

Impact of Action Space. We assess the impact of the number of action phrases for building the common action space, as shown in Figure 7(a). In extreme cases, if there exists no action space and we directly learn the customized token embedding from the action encoder (w/o action space), the model can not generate customized actions and is overfitted to the input image details, such as the actor's clothes and the background. When utilizing a smaller number of action phrases (w/ 50% action phrases), the generated quality is not as good as TwinAct.

Impact of Action Encoder and Action Similarity Loss. We assess the impact of the action encoder and action similarity loss in Figure 7(b) and Table 2. The experimental results reveal that employing an action encoder can enhance discriminative capabilities regarding actions and yield satisfactory outcomes. In contrast, without an action encoder, optimizing the coefficients W_k of the action bases B_k through back-propagation (w/o action encoder) is ineffective. Furthermore, we can find action similarity loss can guide model towards more accurately imitating the customized actions by drifting the model's focus from low-level pixel features to action-related high-level semantic features.

Impact of the Number of Principal Components. We explore the effect of the number of principal components p on model performance. With 768 dimensions in the CLIP text embeddings, we conduct experiments with different values of p from $\{64, 256, 512, 768\}$. The experimental results, outlined in Table 2, indicate that the optimal results are achieved when p is set to 512. Additionally, we showcase the variations in the generated images for different p values in Figure 7(c).

4.4 Combinations of Customized Actors and Customized Actions

Our method demonstrates strong adaptability to integrate with existing subject-driven customized generation methods[11], as shown in Figure 5. It effectively adapts to customized actors, which were unknown by TGDMs, and generates visually engaging images showcasing these actors performing customized actions. In contrast, as depicted in Figure 10 in Appendix, existing methods struggle to accurately capture action-related features, resulting in unsatisfactory generation of customized actions and a lack of identity consistency for the customized actors.

https://doi.org/10.52202/079017-0390

5 Conclusion

In this paper, we introduce a novel approach, TwinAct, for customized action image generation, which aims to preserve the fidelity of the action and the consistency of the actor's identity. TwinAct consists of a common action space and can create new actions by adjusting or combining the action bases in the space. By imitating action via action bases, TwinAct effectively decouples action from the actor. Extensive experiments demonstrate TwinAct's superiority in generating accurate, context-independent customized actions across various subjects, including humans, animals, and customized actors. Furthermore, TwinAct's ability to preserve the high fidelity and identity consistency in generated images highlights its robustness and adaptability. The potential applications of TwinAct in areas such as animation, gaming, and visual effects are vast, offering new opportunities for creative and personalized content generation.

6 Acknowledgements

This work was supported by the National Natural Science Foundation of China (No.62307032, No.62037001) and the Key Research and Development Program of Zhejiang Province (No. 2023C03192).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 157–164. 2023.

[3] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[4] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

[5] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Encoder-based domain tuning for fast personalization of text-to-image models. *ACM Transactions on Graphics (TOG)*, 42(4):1–13, 2023.

[6] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[7] Siteng Huang, Biao Gong, Yutong Feng, Xi Chen, Yuqian Fu, Yu Liu, and Donglin Wang. Learning disentangled identifiers for action-customized text-to-image generation. *arXiv preprint arXiv:2311.15841*, 2023.

[8] Ziqi Huang, Tianxing Wu, Yuming Jiang, Kelvin CK Chan, and Ziwei Liu. Reversion: Diffusion-based relation inversion from images. *arXiv preprint arXiv:2303.13495*, 2023.

[9] LIII KPFRS. On lines and planes of closest fit to systems of points in space. In *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems (SIGMOD)*, page 19, 1901.

[10] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023.

[11] Wang Lin, Jingyuan Chen, Jiaxin Shi, Yichen Zhu, Chen Liang, Junzhong Miao, Tao Jin, Zhou Zhao, Fei Wu, Shuicheng Yan, et al. Non-confusing generation of customized concepts in diffusion models. *arXiv preprint arXiv:2405.06914*, 2024.

https://doi.org/10.52202/079017-0390

[12] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[13] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[14] Saman Motamed, Danda Pani Paudel, and Luc Van Gool. Lego: Learning to disentangle and invert concepts beyond object appearance in text-to-image diffusion models. *arXiv preprint arXiv:2311.13833*, 2023.

[15] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[16] Anton Pelykh, Ozge Mercanoglu Sincan, and Richard Bowden. Giving a hand to diffusion models: a two-stage approach to improving conditional human image generation. *arXiv preprint arXiv:2403.10731*, 2024.

[17] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[18] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[19] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[20] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[21] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[22] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023.

[23] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[24] Nahian Siddique, Sidike Paheding, Colin P Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *Ieee Access*, 9:82031–82057, 2021.

[25] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[26] Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc N Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2116–2127, 2023.

[27] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023.

https://doi.org/10.52202/079017-0390

[28] Chengrui Wang, Pengfei Liu, Min Zhou, Ming Zeng, Xubin Li, Tiezheng Ge, et al. Rhands: Refining malformed hands for generated images with decoupled structure and style guidance. *arXiv preprint arXiv:2404.13984*, 2024.

[29] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024.

[30] Ge Yuan, Xiaodong Cun, Yong Zhang, Maomao Li, Chenyang Qi, Xintao Wang, Ying Shan, and Huicheng Zheng. Inserting anybody in diffusion models via celeb basis. *arXiv preprint arXiv:2306.00926*, 2023.

[31] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

https://doi.org/10.52202/079017-0390

In this appendix, we first provide an in-depth analysis of Stable Diffusionin in Section A.1 and ControlNet in Section A.3, and additional results of TwinAct in Section A.3 and A.4, demonstrating its effectiveness in generating customized action images. Then we introduce the evaluation setting in Section B.1, the details of the action space in Section B.2, and the implementation details in Section C. Finally, we discuss the limitations in Section D and societal implications in Section E for TwinAct.

A More Qualitative Results

A.1 Comparison with Stable Diffusion

We first utilize GPT-4 to generate a detailed description of the actions in the sample image provided by the user, as shown in Figure 12. We can find that on the one hand, although GPT-4's description is highly sufficiently, it still inevitably loses many details of the action. On the other hand, existing text-guided image diffusion generation models have difficulty in understanding such detailed textual descriptions, like "heart" in the fourth row in Figure 12. Both reasons result in the generation of action images that are far from the sample image. In contrast, our approach accurately captures action-related features, and the user only needs to provide textual prompts like "a polar bear V^*", and then customized action images can be generated in the desired contexts.

A.2 Comparison with ControlNet

In addition to controlling the generation of models through textual conditioning, current work such as ControlNet supports extracting spatial constraints from reference images as conditions to generate images. We tried two reference images, skeleton images, and sketches, as conditions to control the generative model to generate customized action images. The results are shown in Figure 9. We can find that the skeleton image-based approach loses a lot of details of the action, such as fingers, and it is difficult to imitate the customized action. The other sketch-based approach provides more details, but this limits the diversity of the model generation results, e.g., it is difficult to generate images of animals performing customized actions since the sketches contain human bodies. Also, the sketches contain some additional action-irrelevant information such as the background, the actor's hairstyle, and so on. More importantly, the controlled generation model lacks the understanding of the contextual dependency of the action, as shown in the third row in Figure 9. It is difficult for the generation model to understand the relationship between the hands and the feet in the customized action, which leads to serious body distortions and visual deficiencies, and this becomes even more obvious when animals are involved.

A.3 Comparison with Baseline on Customized Actor

In this section, we compare the results of existing customized TGDMs and TwinAct in generating images containing customized actors and customized actions. As shown in Figure 10, our approach significantly outperforms the previous methods in maintaining the fidelity of the action and consistency of the character's identity. Textual Inversion, Dreambooth, and Revrsion struggle to imitate the customized action. As for, Custom Diffusion, P+, and ADI, although they can roughly maintain the consistency of the actions specified in the sample image, fail to maintain the identity information of the customized actors. This is because their customized action tokens are confounded with information irrelevant to the actions in the sample image, *e.g., Black Widow*'s red curly hair in the first row, and the yoga pants in the second row.

A.4 Additional Results

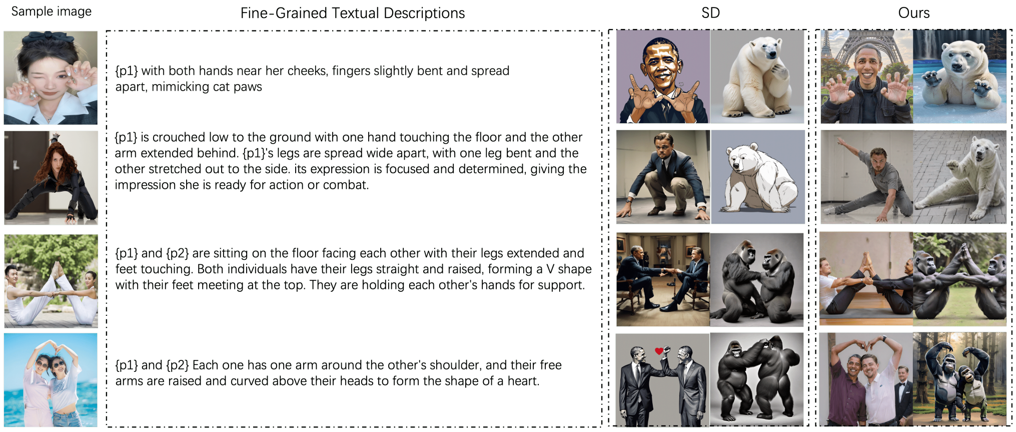
In this section, we show more customized action images generated by TwinAct, as shown in Figure11, covering the rest of the 5 actions within our benchmark.

B Experiment Design

B.1 Evaluation Setting

Following [7], we provide 25 subjects for evaluation as follows:

https://doi.org/10.52202/079017-0390



{p1} with both hands near her cheeks, fingers slightly bent and spread apart, mimicking cat paws

{p1} is crouched low to the ground with one hand touching the floor and the other arm extended behind. {p1}'s legs are spread wide apart, with one leg bent and the other stretched out to the side. its expression is focused and determined, giving the impression she is ready for action or combat.

{p1} and {p2} are sitting on the floor facing each other with their legs extended and feet touching. Both individuals have their legs straight and raised, forming a V shape with their feet meeting at the top. They are holding each other's hands for support.

{p1} and {p2} Each one has one arm around the other's shoulder, and their free arms are raised and curved above their heads to form the shape of a heart.

Figure 8: **Comparing the results generated by TwinAct and fine-grained textual descriptions.** The results show that customized actions are difficult to describe and that existing text-to-image generation models do not accurately follow textual instructions even when fine-grained textual descriptions are provided.

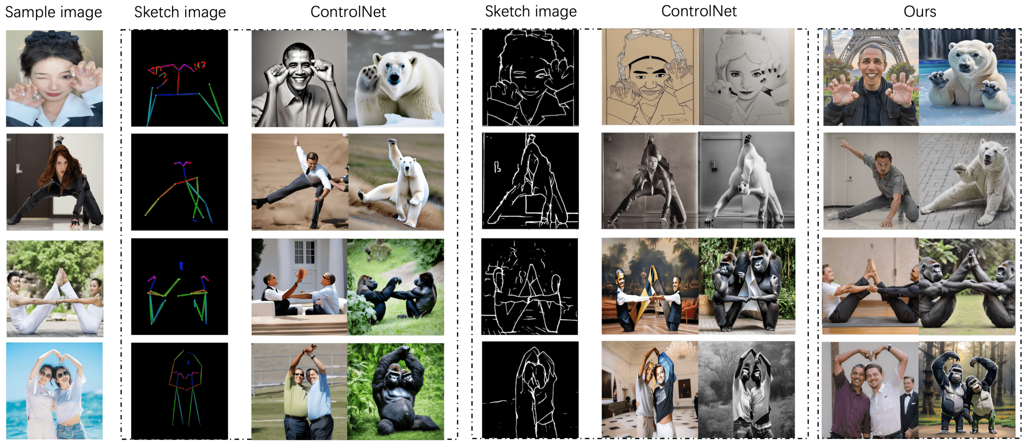


Figure 9: **Comparison of results generated by TwinAct and sketch-based generation models.** The results show that it is difficult for the user to provide a suitable sketch for generating customized action images. The results generated with skeleton images show it is difficult to capture the details of the action such as fingers, while the results generated with line images are limited in generalization, especially when it involves animals.

- **generic human:** “A boy”, “A girl”, “A man”, “A woman”, “An old man”
- **well-known personalities:** “Barack Obama”, “Michael Jackson”, “David Beckham”, “Leonardo DiCaprio”, “Messi”, “Spiderman”, “Batman”
- **animals:** “A dog”, “A cat”, “A lion”, “A tiger”, “A bear”, “A polar bear”, “A fox”, “A cheetah”, “A monkey”, “A gorilla”, “A panda”
- **customized actors:** “Cruze”, “Imelda”

where the inclusion of diverse and previously unseen actors, as well as animals and customized characters, requires models to retain pre-trained knowledge while also generating accurate and undistorted representations of these actors.

B.2 Action phrases for Action Sapce

As stated in the manuscript, we devise a set of action phrases to build action space. Specifically, we collect a comprehensive list of about 1,200 action phrases with the help of GPT-4 [1]. After obtaining these action phrases, in order to filter out those action phrases that are not recognized by the TGDMs.

https://doi.org/10.52202/079017-0390

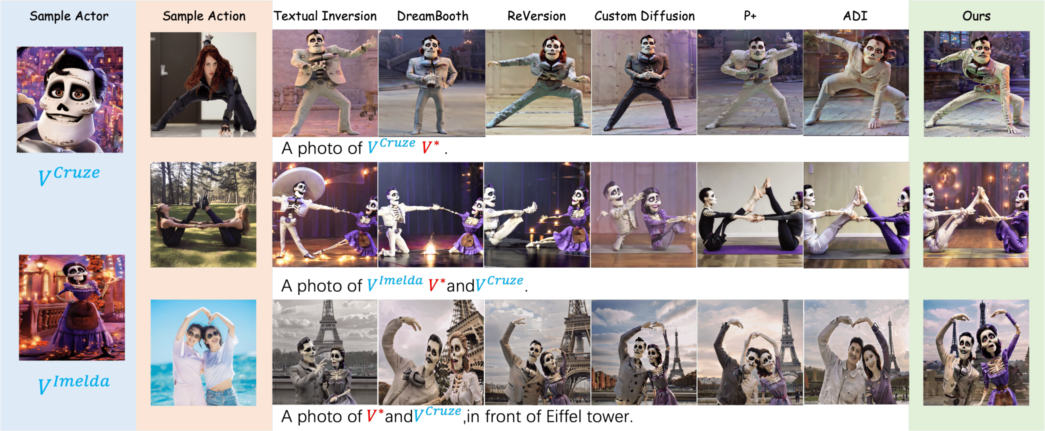

Sample Actor
Sample Action
Textual Inversion
DreamBooth
ReVersion
Custom Diffusion
P+
ADI
Ours
V^{Cruze}
A photo of V^{Cruze} V^* .
A photo of V^{Imelda} V^*andV^{Cruze}.
V^{Imelda}
A photo of V^*andV^{Cruze},in front of Eiffel tower.

Figure 10: **Comparing the results of generating customized actors to perform customized actions generated by TwinAct and other methods.** TwinAct can better maintain the identity consistency of the subject in the generated image, and the fidelity of the action.

we synthesize the image result by providing 2 types of prompts, *i.e.* "A photo of *{Action}*", "A photo of someone *{Action}*", to the Stable Diffusion XL [17], where *{Action}* represents an action phrases (*e.g.* "raise hand"). Those action phrases that do not generate the correct action, such as *"reverse plank"*, or *"crab walk"*, are filtered out. We ended up with 832 valid common action phrases, which were then further tokenized and encoded into the action space.

C More Implementation Deatils

C.1 Implementation Details about TwinAct

Implementation Details. For our method, we use the AdamW[12] optimizer with a learning rate of 2e-4. We use CLIP as a preprocessor to estimate the action of the given reference image. Unless otherwise specified, Stable Diffusion XL is selected as the default pre-trained model, and images are generated at a resolution of 1024×1024. All experiments are conducted on A-100 GPUs. We fine-tuning text embedding along with the LoRA layer. We integrate the LoRA layer into the linear layer within all attention modules of the U-net, utilizing a rank of $r = 8$.

Sample Details. All experiments and evaluations make use of the DDPM [25] with 50 sampling steps with a scale of 7.5 for all methods. To ensure consistency and filter out undesired variations in diffusion models, we follow the approach outlined in [11] by employing the same negative prompt for both our method and the comparison methods during sampling. The negative prompt used is "long body, low-res, bad anatomy, bad hands, missing fingers, extra digit, fewer digits, cropped, worst quality, low quality."

Running Times. The process of tuning a customized action token embedding in the text encoder and the LoRA layer in the text-to-image decoder typically requires approximately 15 minutes using an Nvidia-A100 GPU.

C.2 Implementation TwinAct with Customized Actor

Thanks to the rapid development of subject-driven customized text-guided diffusion models for image generation, they achieved an exciting proceeding in generating high-fidelity customized actor images. We combine TwinAct with the existing state-of-the-art subject-driven customized TGDMs [11] to generate images of customized actors performing customized actions. To achieve this, we only need to apply the action base alone to the embedding of the customized action tokens, while the embedding of the customized actor's tokens is obtained as in [11]. The action similarity loss can be applied to the optimization of both the customized action tokens and the customized actor tokens, and our generation results show that the action similarity loss does not affect the learning of the customized actor tokens.

https://doi.org/10.52202/079017-0390

D Limitation and Future Work

In this work, our goal is to generate images of customized actions. There are two limitations to TwinAct. First, TwinAct obtains the embedding of customized action toknes by combining action bases. Although the action base is carefully defined, it is still possible to miss some actions, which limit the expressiveness of TwinAct's output domain. This can be overcome by further expanding the action space by adding more common action phrases. Another limitation comes from the physical defects of TGDMs in generating human bodies, we mitigate this problem by providing some negative prompts like "long body, low-res, bad anatomy, bad hands, missing fingers, extra digit, fewer digits, cropped, worst quality, low quality". Furthermore, there are some works [29, 16, 28] designed to solve the problem. Moreover, TwinAct is not restricted to image generation models; it can also be used for customizing text-guided diffusion models to generate videos or 3D assets. we leave it as future work.

E Expected Societal Implications

We aim to address the challenge of decoupling actions and actors to customize the text-guided diffusion model for few-shot action image generation. A primary ethical concern is the potential misuse of this technology, notably in creating deepfakes, which can result in misinformation, privacy violations, and other harmful outcomes. To mitigate these risks, the establishment of robust ethical guidelines and continuous monitoring is essential.

The concern raised here is a common one, not just for our method but across various multi-concept customization techniques. A viable strategy to lessen these risks might be the implementation of tactics akin to those used in anti-dreambooth[26]. This approach involves adding minor noise disturbances to the shared images, thereby hindering the customization process. Furthermore, embedding invisible watermarks in the generated images can serve as a deterrent against misuse and ensure that they are not used without due acknowledgment.

https://doi.org/10.52202/079017-0390

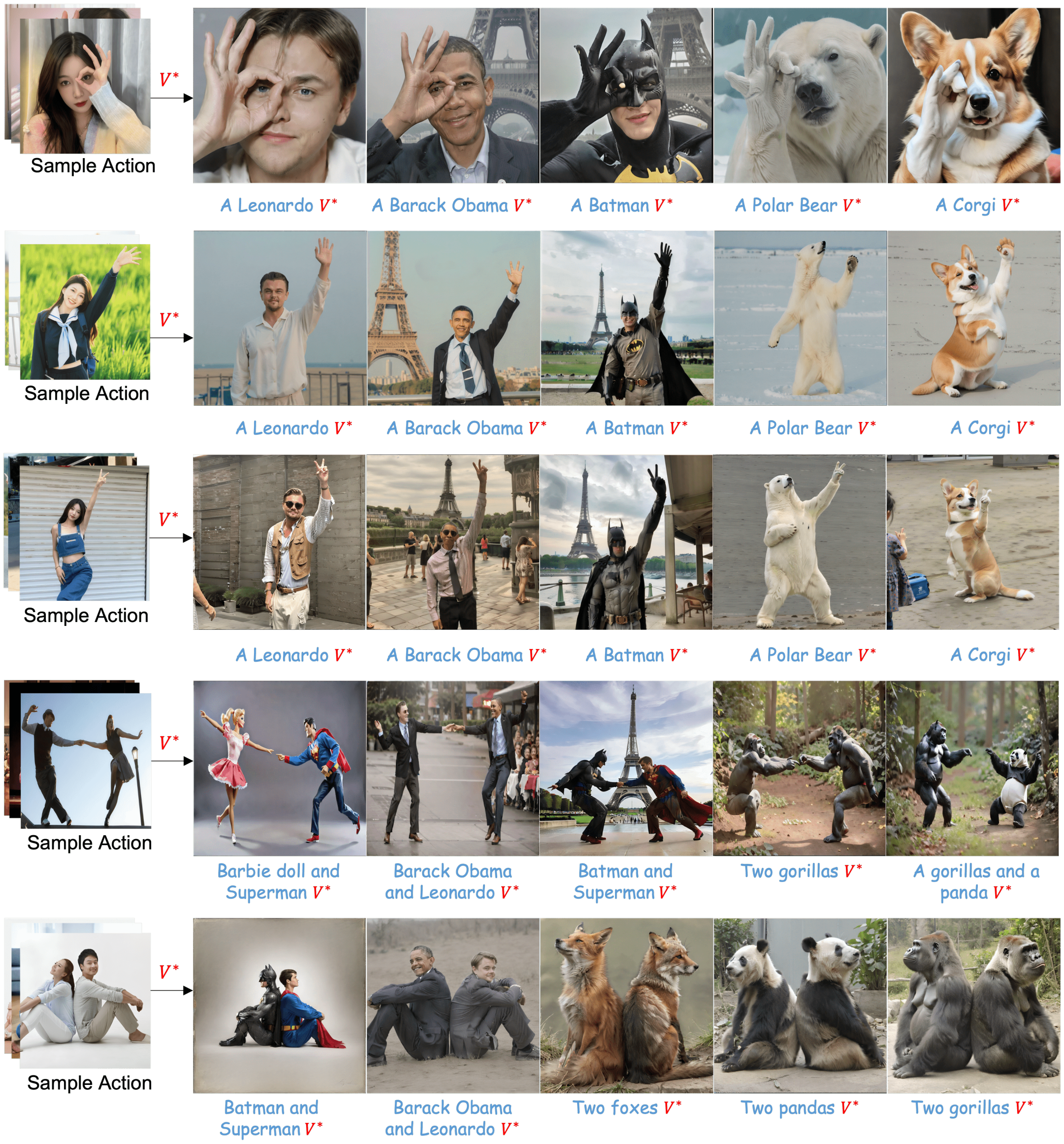


Figure 11: **More TwinAct generated results.**

Hands

Hand on shoulder
Waving goodbye
Slapping forehead
Covering mouth
Hand on heart
Casting spell
Raising hand
Folding arms
Crossing arms
Pointing forward
Punching air
Hugging self
Pushing away
Pulling rope
Lifting weights
Clapping hands
Shaking hands
Patting back
Swinging arm
Stretching arm
Waving hand
Shaking fist
Brushing hair
Scratching head
Lifting arm
Grabbing shoulder
Reaching out
Twisting arm
Flexing wrist
Flipping hand
Pulling back arm
Swinging arm forward
Swinging arm backward
Throwing arm forward
Pulling arm up
Pushing arm down
Circling arm
Shielding with arm
Lowering hand
Shaking arm
Resting arm on hip
Leaning on elbow
Draping arm over head
Tucking arm behind back
Swinging arm side to side
Lifting arm over head
Lowering arm to side
Extending arm forward
Resting arm on knee

Finger

Fingers crossed
Finger snapping
Thumbs up
Spinning finger
Finger gun
Tapping fingers
Fist bump
Palm up asking
Praying hands
Knuckle cracking
Hand on shoulder
Peace sign
Pinching gesture
Wrist flex
Slapping forehead
Grabbing wrist
Finger circling
Clenched fist
Open palm
Pointing down
Thumbs down
Palm facing down
Finger tips touch
Holding object
Boxing punch
Counting fingers
Wagging finger
Wiping tears
Typing gesture
Finger pointing up
Finger pointing sideways
Finger pinching
Thumb flexing
Finger curling
Finger pressing
Thumb pressing
Finger rubbing
Finger crossing over
Finger crossing under
Finger forming heart
Finger forming circle
Thumb pointing left
Thumb pointing right
Thumb sliding down
Thumb sliding up
Finger forming V
Make a fist
Put palms together
Finger snap

Heads

Nodding head yes
Shaking head no
Tilting head sideways
Looking up
Looking down
Turning head left
Turning head right
Tucking chin
Raising eyebrows
Furrowing brows
Blinking eyes
Winking eye
Squinting eyes
Gazing forward
Glancing sideways
Staring intently
Glaring angrily
Smiling broadly
Frowning deeply
Puckering lips
Biting lip
Pressing lips
Opening mouth
Closing mouth
Yawning widely
Chewing food
Gritting teeth
Pouting lips
Sticking tongue
Licking lips
Tilting head up
Tilting head down
Tilting head left
Tilting head right
Rotating head clockwise
Jerking head back
Jerking head forward
Jerking head sideways
Scratching chin
Puffing cheeks
Gnashing teeth
Stroking chin
Lowering eyelids
Sucking on a tooth
Pinching nose
Widening eyes
Narrowing eyes
Rubbing nose
Rubbing eyes

Full-Body

Crouching stealthily
Sprawling out
Pouncing forward
Embracing warmly
Shoving forcefully
Expanding chest
Contracting abdomen
Twirling gracefully
Pirouetting steadily
Bearing down
Leaping high up
Ducking quickly down
Tumbling forward roll
Flipping backward somersault
Shielding face hands
Kneeling on ground
Squatting to rest
Push-up
Plank
Sit-up
Crunch
Burpee
Squat
Lunge
Deadlift
High knees
Butt kick
Handstand
Arching body backward
Curling body forward
Leg swings
Leg lifts
Leg drops
Bicycle kicks
Flutter kicks
Crisscross kicks
Scissor kicks
Side kicks
Front kicks
Back kicks
Inward heel taps
Outward heel taps
Toe touches
Walking
Running
Jogging
Sprinting
Skipping
Jumping

Figure 12: **Some examples of action phrases.**

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In Section 1

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Appendix

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

https://doi.org/10.52202/079017-0390

Justification: In Section 3

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In Section 4 and Appendix

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

https://doi.org/10.52202/079017-0390

Answer: [Yes]

Justification: In Appendix

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: In Section 4.1

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In Section 4.2

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

https://doi.org/10.52202/079017-0390

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In Section 4

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: In Appendix

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: In Appendix

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

https://doi.org/10.52202/079017-0390

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: N/A

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: In Reference

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

https://doi.org/10.52202/079017-0390

Answer: [NA]

Justification: N/A

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: In Appendix

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: N/A

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-0390