ELITE: Encoding Visual Concepts into Textual Embeddings for Customized Text-to-Image Generation

Yuxiang Wei[1,2] Yabo Zhang[1] Zhilong Ji[3] Jinfeng Bai[3] Lei Zhang[2] Wangmeng Zuo[1,4(✉)]
[1]Harbin Institute of Technology [2] The Hong Kong Polytechnic University [3]Tomorrow Advancing Life [4] Peng Cheng Lab

Abstract

In addition to the unprecedented ability in imaginary creation, large text-to-image models are expected to take customized concepts in image generation. Existing works generally learn such concepts in an optimization-based manner, yet bringing excessive computation or memory burden. In this paper, we instead propose a learning-based encoder, which consists of a global and a local mapping networks for fast and accurate customized text-to-image generation. In specific, the global mapping network projects the hierarchical features of a given image into multiple "new" words in the textual word embedding space, i.e., one primary word for well-editable concept and other auxiliary words to exclude irrelevant disturbances (e.g., background). In the meantime, a local mapping network injects the encoded patch features into cross attention layers to provide omitted details, without sacrificing the editability of primary concepts. We compare our method with existing optimization-based approaches on a variety of user-defined concepts, and demonstrate that our method enables high-fidelity inversion and more robust editability with a significantly faster encoding process. Our code is publicly available at https://github.com/csyxwei/ELITE.

1. Introduction

Recently, large-scale diffusion models [3, 29, 32, 34] have demonstrated impressive superiority in text-to-image generation. By training with billions of image-text pairs, large text-to-image diffusion models have exhibited excellent semantic understanding ability, and generate diverse and photo-realistic images being accordant to the given text prompts. Owning to their unprecedentedly creative capabilities, these models have been applied to various tasks, such as image editing [16, 23], data augmentation [24], and even artistic creation [26].

However, despite diverse and general generation, users may expect to create imaginary instantiates with undescribable personalized concepts [18], *e.g.*, "corgi" in Fig. 1.

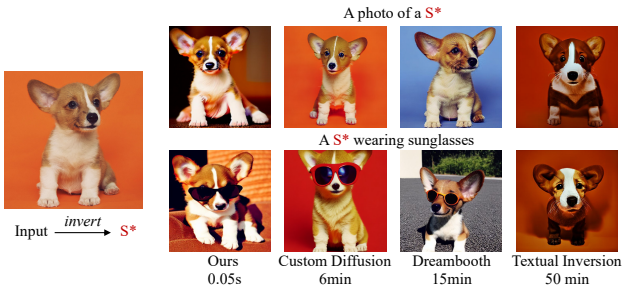


Figure 1. Given an input image, customized text-to-image generation learns a pseudo-word (S*) in word embedding space to represent the target concept. With S*, one can flexibly synthesize or edit the concept with text prompts. The running time to learn a new concept is also listed, and our method learns the new concept much faster than others.

To this end, many recent studies have been conducted for customized text-to-image generation [9, 15, 18, 33], which aims to learn a specific concept from a small set of user-provided images (*e.g.*, 3∼5 images). Then, users can flexibly compose the learned concepts into new scenes, *e.g.*, `A S* wearing sunglasses` in Fig. 1. Given a small image set depicting the target concept, Textual Inversion [15] learned a new pseudo-word (*i.e.*, S*) in the well-editable textual word embedding space of text encoder to represent the user-defined concept. DreamBooth [33] finetuned the entire diffusion model to accurately align the target concept with a unique identifier. Custom Diffusion [18] balanced the fidelity and memory by selectively finetuning K, V mapping parameters in cross attention layers.

Albeit flexible generation has been achieved by [15, 18, 33], the computational efficiency remains a challenge to obtain the textual embedding of a visual concept. Existing methods usually adopt the per-concept optimization formulation, which requires several or tens of minutes to learn a single concept. As shown in Fig. 1, the Custom Diffusion [18], which is among the fastest existing algorithms, still takes around 6 minutes to learn one concept, which is infeasible for online applications. In contrast, in GAN inversion, many efficient learning-based methods [31] have

been proposed to accelerate the optimization process. An encoder can be trained to infer the latent codes, which only needs one step forward inference.

Driven by the above analysis, we propose a learning-based encoder for **E**ncoding visua**L** concepts **I**nto **T**extual **E**mbeddings, termed as **ELITE**. As shown in Fig. 2, our ELITE adopts a pre-trained CLIP image encoder [28] for feature extraction, followed by a global mapping network and a local mapping network to encode visual concepts into textual embeddings. Firstly, we train a global mapping network to map the CLIP image features into the textual word embedding space of the CLIP text encoder, which has superior editing capacity [15]. Since a given image contains both the subject and irrelevant disturbances, encoding them as a single word embedding severely degrades the editability of subject concept. Thus, we propose to separately learn them with a well-editable primary word and several auxiliary words. Using the hierarchical features from CLIP intermediate layers, the word learned from the deepest features naturally links to the primary concept (*i.e.*, the subject), while auxiliary words learned from other features describe the irrelevant disturbances (as shown in Fig. 5). When deploying to customized generation, we only use the primary word to avoid editability degradation from auxiliary words.

Usually, a visual concept is worth more than one word, and describing it with a single word may result in the inconsistency of local details [15]. For higher fidelity of the learned concept without sacrificing its editability, we further propose a local mapping network to inject finer details. From Fig. 2, our local mapping network encodes the CLIP features into the textual feature space (*i.e.*, the output space of the text encoder). Compared with the textual word embeddings learned by global mapping network, the textual feature embeddings focus on the local details of each patch in the given image. Then, the obtained textual feature embeddings are injected through additional cross attention layers, and the output feature is fused with the global part to improve the local details. Experiments show that our ELITE can encode the target concept efficiently and faithfully, while keeping control and editing abilities. The contributions of this work are summarized as follows:

- We propose a learning-based encoder, namely ELITE, for fast and accurate customized text-to-image generation. It adopts a global and a local mapping networks to encode visual concepts into textual embeddings.
- Multi-layer features are adopted in global mapping to learn a well-editable primary word embedding, while the local mapping improves the consistency of details without sacrificing editability.
- Experimental results show that our ELITE can faithfully recover the target concept with higher visual fidelity, and enable more robust editing.

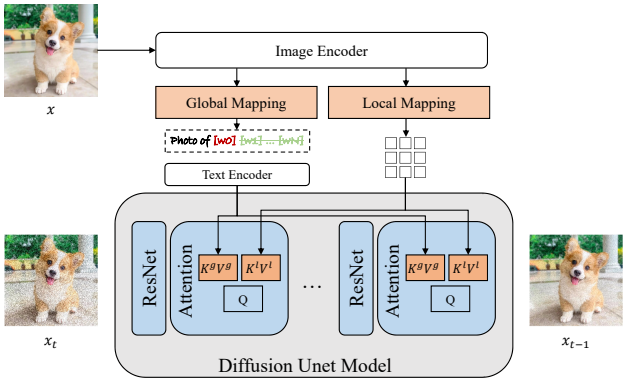


Figure 2. **Inference pipeline of our proposed ELITE**. Given a user-provided image x, our ELITE extracts hierarchical features with CLIP image encoder. Then, it uses global and local mapping networks to encode the visual concept into textual word embeddings (*i.e.*, primary word w_0 and auxiliary words $w_{1\ldots N}$) and textual feature embeddings, respectively. The embeddings are injected with cross attention to guide customized generation. Note that, in generation, only w_0 is used for better editability.

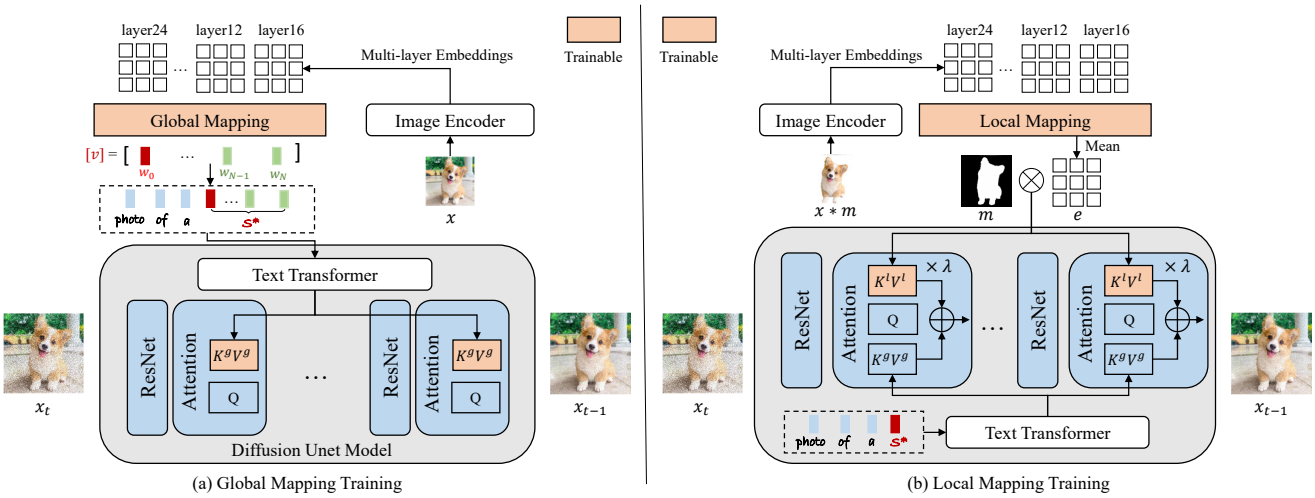
2. Related Work

2.1. Text-to-Image Generation

Deep generative models have achieved tremendous success on text-conditioned image generation [3, 6, 11, 12, 14, 20, 25, 29, 30, 32, 34, 35, 38, 40] and have recently attracted intensive attention. They can be categorized into three groups: GAN-based, VAE-based, and diffusion-based models. Albeit GAN-based [20, 35, 38] and VAE-based models [6, 11, 14, 30, 40] can synthesize images with promising quality and diversity, they still cannot match user descriptions very well. Recently, diffusion models have shown unprecedentedly high-quality and controllable imaginary generation and been broadly applied to text-to-image generation [3, 12, 25, 29, 32, 34]. By training with massive corpora, these large text-to-image diffusion models, such as DALLE-2 [29], Imagen [34], and Stable Diffusion [32] have demonstrated excellent semantic understanding, and can generate diverse and photo-realistic images according to a given text prompt. However, despite the superior performance on general synthesis, they still struggle to express the specific or user-defined concepts, *e.g.*, "corgi" in Fig. 1. Our method focuses on making pre-trained diffusion models to learn these new concepts efficiently.

2.2. GAN Inversion

GAN inversion refers to projecting real images into latent codes so that images can be faithfully reconstructed and edited with pre-trained GAN models [22, 39]. Generally speaking, there are two types of GAN inversion algorithms in the literature: i) *optimization-based:* directly optimize latent code to minimize the reconstruction error [4, 10, 21], and ii) *encoder-based:* train an encoder to invert an image



(a) Global Mapping Training (b) Local Mapping Training

Figure 3. **Training pipeline of our proposed ELITE**. Our method consists of two stages: **(a)** a global mapping network is first trained to encode a concept image into multiple textual word embeddings, with one primary word (w_0) for well-editable concept and other auxiliary words ($w_{1...N}$) to exclude irrelevant disturbances. **(b)** A local mapping network is further trained, which projects the foreground object into textual feature space to provide local details. In this stage, only the well-editable primary word (w_0) is used.

into latent space [1, 2, 27, 31, 37]. The optimization-based methods [4, 10, 21] usually require hundreds of iterations to obtain promising results, while encoder-based methods greatly accelerate this process via one feed-forward pass only. To improve image fidelity without compromising editability, HFGI [37] embeds the omitted information into high-rate features. Similarly, our ELITE adopts a local mapping network that encodes the concept images into textual feature space to improve details consistency.

2.3. Diffusion-based Inversion

The inversion of text-to-image diffusion models can be performed in two types of latent spaces: the Textual Word Embedding (TWE) space [13, 15, 18, 33] of text encoder or the Image-based Noise Map (INM) space [8, 23, 36]. INM-based inversion methods, such as DDIM [36] or Null Text [23] find the initial noise to reconstruct image faithfully, while it suffers from a degraded editing ability. In contrast, the TWE space has shown superior editing capacity, which is well suitable for textual inversion and customized generation [13, 15, 18]. For example, Textual Inversion [15] and DreamArtist [13] optimize the embedding of new "words" using a few user-provided images to recover the target concept. DreamBooth [33] finetunes the entire text-to-image model to learn high-fidelity new concept with a unique identifier. To improve the computation efficiency, Custom Diffusion [18] only updates the key and value mapping parameters in cross attention layers with better performance. Albeit flexible generation has been achieved, existing methods still require several or tens of minutes to learn a single concept.

In this work, we choose the TWE space as target for inversion, while proposing a learning-based encoder ELITE for fast and accurate customized text-to-image generation. With the proposed global and local mapping networks, our method can learn the new concept quickly and faithfully with one single image.

3. Proposed Method

Given a pretrained text-to-image model ϵ_θ and an image x indicating the target concept (usually an object), customized text-to-image generation aims to learn a pseudo-word (S*) in word embedding space to describe the concept faithfully, while keeping the editability. To achieve fast and accurate customized text-to-image generation, we propose an encoder ELITE to encode the visual concept into textual embeddings. As illustrated in Fig. 2, our ELITE first adopts a global mapping network to encode visual concepts into the textual word embedding space. The obtained word embeddings can be composed with texts flexibly for customized generation. To address the information loss in word embedding, we further propose a local mapping network to encode the visual concept into textual feature space to improve the consistency of the local details. In the following, we begin by presenting an overview of the text-to-image model utilized in our approach (Sec. 3.1). Then, we will introduce the details of the proposed global mapping network (Sec. 3.2) and local mapping network(Sec. 3.3).

3.1. Preliminary

In this work, we employ the Stable Diffusion [32] as our text-to-image model, which is trained on large-scale data and consists of two components. First, the autoencoder ($\mathcal{E}(\cdot)$, $\mathcal{D}(\cdot)$) is trained to map an image x to a lower dimensional latent space by the encoder $z = \mathcal{E}(x)$. While the decoder $\mathcal{D}(\cdot)$ learns to map the latent code back to the image so that $D(\mathcal{E}(x)) \approx x$. Then, the conditional diffu-

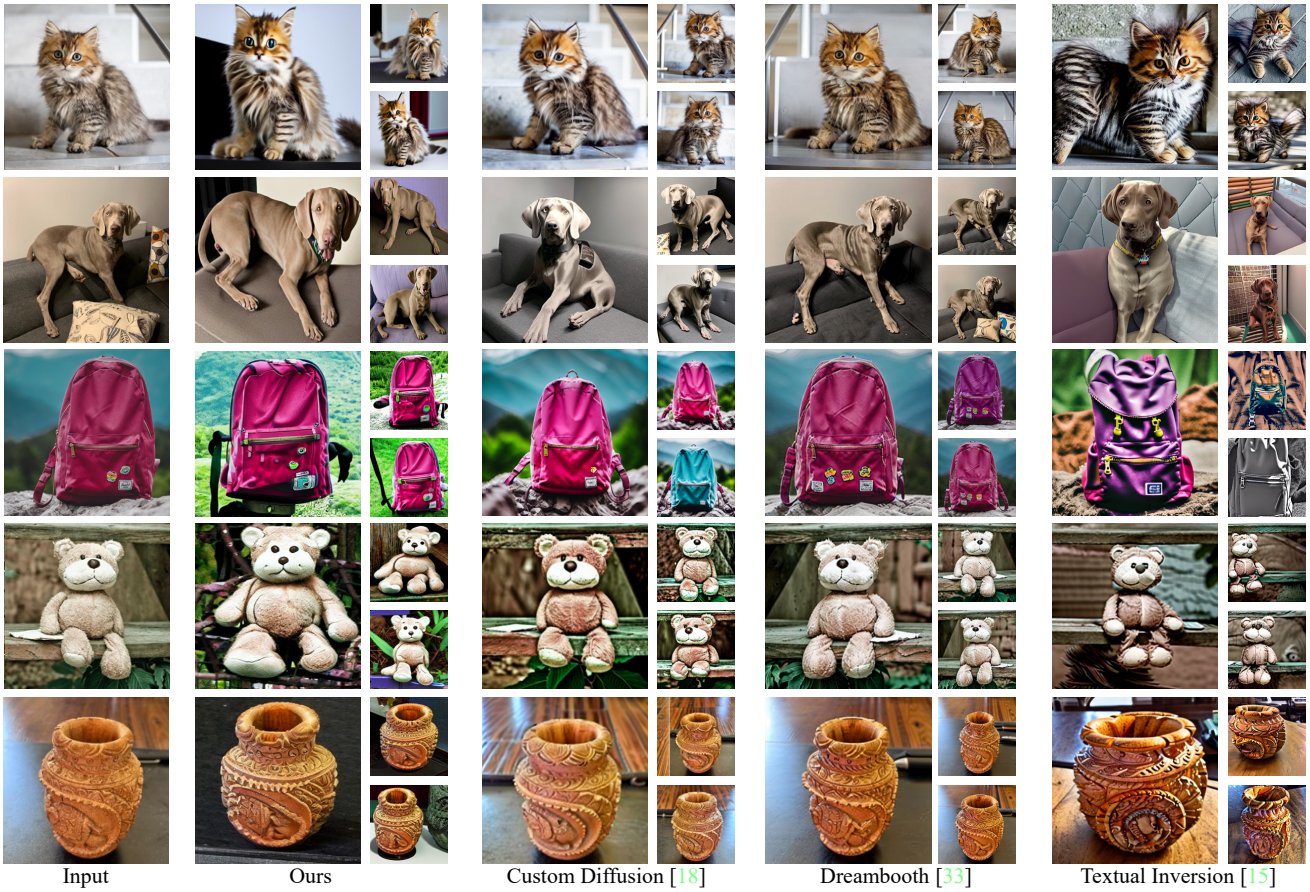


Figure 4. **Visual comparisons of concept generation.** For generating images, `A photo of a S*` is used for Textual Inversion [15] and our method, while `A photo of a S* [category]` is used for Dreambooth [33] and Custom Diffusion [18]. Our ELITE shows comparable performance to it competitors.

sion model $\epsilon_\theta(\cdot)$ is trained on the latent space to generate latent codes based on text condition y. We simply adopt the mean-squared loss to train the diffusion model:

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

where ϵ denotes the unscaled noise, t is the time step, z_t is the latent noise at time t, and $\tau_\theta(\cdot)$ represents the pretrained CLIP text encoder [28]. During inference, a random Gaussian noise z_T is iteratively denoised to z_0, and the final image is obtained through the decoder $x' = \mathcal{D}(z_0)$.

To incorporate text information in the process of image generation, cross attention is adopted in Stable Diffusion. Specifically, the latent image feature f and text feature $\tau_\theta(y)$ are first transformed by the projection layers to obtain the query $Q = W_Q \cdot f$, key $K = W_K \cdot \tau_\theta(y)$ and value $V = W_V \cdot \tau_\theta(y)$. W_Q, W_K, and W_V are weight parameters of query, key, and value projection layers, respectively. Attention is conducted by a weighted sum over value features:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d'}}\right)V, \quad (2)$$

where d' is the output dimension of key and query features. The latent image feature is then updated with the output of the attention block.

3.2. Global Mapping

Following [15, 18], we choose the textual word embedding space of CLIP text encoder as the target for inversion. To improve the computation efficiency, we propose a global mapping network that encodes the given concept image into word embeddings directly. As illustrated in Fig. 3(a), to facilitate the embedding learning, the pretrained CLIP image encoder $\psi_\theta(\cdot)$ is adopted as feature extractor, and our global mapping network $M^g(\cdot)$ projects the CLIP features as word embeddings v:

$$v = M^g \circ \psi_\theta(x), \quad (3)$$

where $v \in \mathbb{R}^{N\times d}$, N is the number of words and d is the dimension of word embedding. Global average pooling is employed on features to obtain the word embedding.

Since the image x contains both the desired subject and irrelevant disturbances, encoding them into one word (*i.e.*, $N = 1$) results in an entangled word embedding v with

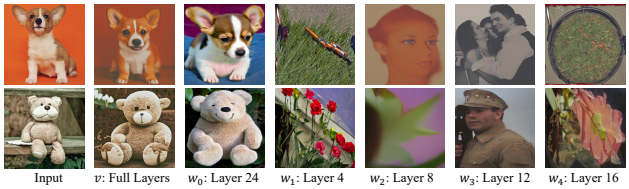


Figure 5. **Visualization of learned word embeddings**. The word associated with the deepest feature (*i.e.*, w_0 from layer 24) describes the primary concept (*i.e.*, corgi, teddybear), while other words describe irrelevant disturbances.

poor editability. To obtain a more informative and editable word embedding, we adopt a multi-layer approach to learn N (N>1) words from the image x separately. Specifically, we select N layers from the CLIP image encoder, and each layer $\psi_\theta^{Li}(\cdot)$ learns one word w_i independently. All words $[w_0, \cdots, w_N]$ are concatenated together to form the textual word embedding v. A pseudo-word S* is further introduced to represent the learned new concept, and v is associated with its word embeddings. To train our global mapping network, we adopt Eqn. (1), and regularize the obtained word embeddings as follows:

$$L_{global} = L_{LDM} + \lambda_{global}\|v\|_1, \tag{4}$$

where λ_{global} is a trade-off hyperparameter. Analogous to [15], we randomly sample a text from the CLIP ImageNet templates [28] as text input during training, such as `a photo of a S*`. The full template list is provided in the *Suppl*. Besides, following [18], the key and value projection layers in the cross attention layer are finetuned with $M^g(\cdot)$, and the obtained new projections are denoted as $K^g = W_K^g \cdot \tau_\theta(y)$ and $V^g = W_V^g \cdot \tau_\theta(y)$.

Benefiting from the hierarchical semantics learned by different layers in the CLIP image encoder, the feature from the deepest layer (*i.e.*, layer 24) possesses the highest comprehension of the image. The word embedding associated with the deepest feature is naturally learned to describe the primary concept (*i.e.*, the subject), while keeping superior editability. In contrast, word embeddings from shallower features are learned to describe the irrelevant disturbances (see Sec. 4.2 for more details). Note that, we use only the word embedding of the deepest feature during local training (Sec. 3.3) and image generation stages for better editability.

3.3. Local Mapping

Usually, a single word embedding is not sufficient to faithfully describe the details of the given concept, while multiple word embeddings may suffer from degraded editing capacity. To improve the consistency between the given concept and synthesized image without sacrificing editability, we further propose a local mapping network. As shown in Fig. 3(b), the local mapping network $M^l(\cdot)$ encodes the multi-layer CLIP features into the textual feature space (*i.e.*, the output space of text encoder):

$$e = M^l \circ \psi_\theta(x * m), \tag{5}$$

where m is the object mask to ease the redundant details of background. $e \in \mathbb{R}^{p\times p\times d}$ keeps the spatial structure and p is the feature size. Each pixel of e focuses on the local details of each patch in the given image. To inject the local information of e into generation, we introduce two additional projection layers W_K^l and W_V^l into cross attention module. The local attention is performed by Attention(Q, K^l, V^l), where $K^l = W_K^l \cdot (e * m)$ and $V^l = W_V^l \cdot (e * m)$. Then, the output feature of local attention is fused with global part to inject finer details:

$$Out = \text{Attention}(Q, K^g, V^g) + \lambda\text{Attention}(Q, K^l, V^l), \tag{6}$$

where λ is a hyperparameter and set as 1 during training. To emphasize more on the object region, the obtained attention map QK^{l^T} is reweighted by $QK^{g^T}_i$, where i is the index of w_0 in the text prompt (more details will be provided in the *Suppl*). To train the local mapping network, we also adopt Eqn. (1) while regularizing the local values V_l:

$$L_{local} = L_{LDM} + \lambda_{local}\|V^l\|_1, \tag{7}$$

where λ_{local} is a trade-off hyperparameter.

Table 1. **Ablation study**. [v] denotes the generated testing results with full word embeddings v, while [w] denotes the generated testing results with the primary word embedding w.

Method	CLIP-T (↑)	CLIP-I (↑)	DINO-I (↑)
Single-Layer Single-Word	0.209	0.714	0.546
Single-Layer Multi-Words [v]	0.198	0.683	0.431
Single-Layer Multi-Words [w]	0.212	0.692	0.443
Multi-Layers Multi-Words [v]	0.204	**0.771**	**0.658**
Multi-Layer Multi-Words [w]	**0.257**	0.699	0.486
Ours w/o Local	**0.257**	0.699	0.486
Ours	0.255	0.762	0.652

4. Experiments

4.1. Experimental Settings

Datasets. To train our local and global mapping networks, we use the `testset` of OpenImages [19] as our training dataset. It contains 125k images with 600 object classes. During training, we crop and resize the object image to 512×512 according to the bounding box annotations. While for local mapping training, mask annotations are also used to extract foreground objects. For customized generation, we adopt concept images from existing works [15, 18, 33] with 20 subjects, including dog, cat, and toy, *etc*. The subject masks can be obtained by a pretrained segmentation model [7, 17, 41]. For quantitative evaluation, we employ the editing prompts from [33], which contains 25 editing prompts for each subject. We randomly generate five images for each subject-prompt pair, obtaining 2,500 images in total. More details can be found in the *Suppl*.

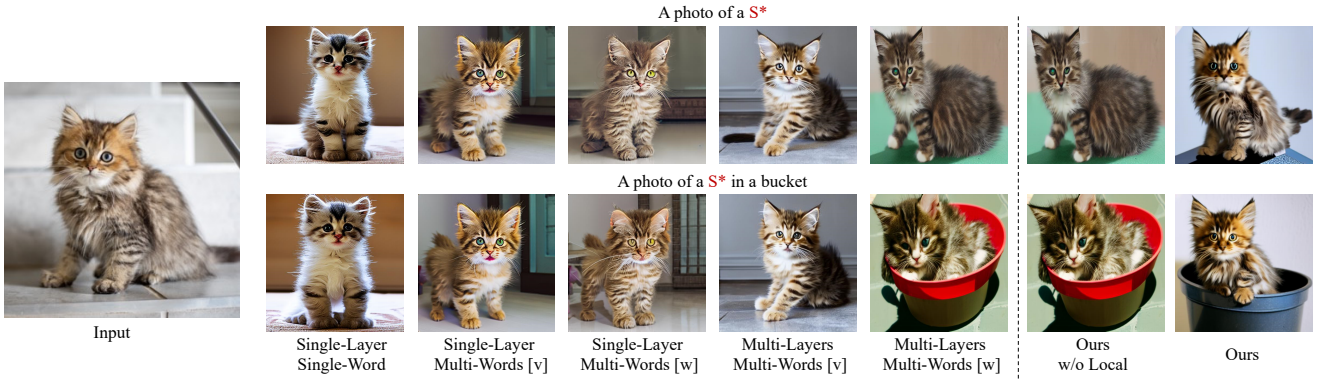


Figure 6. **Visual comparisons of different variants. Left**: [v] denotes the generation results with full word embeddings v, while [w] denotes the generation results with the primary word embedding w. Learning single or multiple word embeddings from the single layer (*i.e*. the deepest layer) fail to achieve reliable editing capacity. In contrast, learning multiple words from multiple layers (*i.e*., our method) successfully learns an editable primary word embedding. **Right**: Our proposed local mapping significantly improves the consistency of details between the input image and the generated image, while maintains editability.

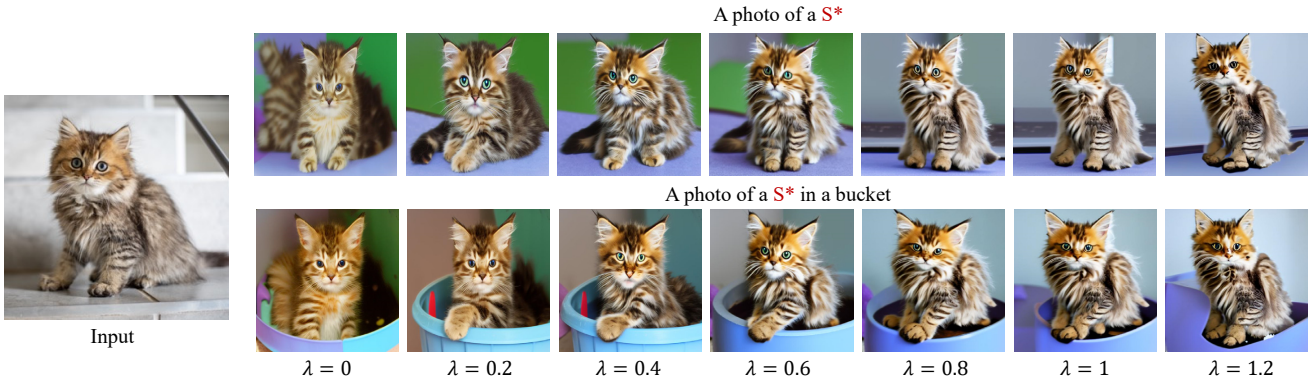


Figure 7. **Visual comparisons by using different values of** λ. As λ increases, the consistency of details between the generated image and the input image improves, yet slightly reducing the editability.

Evaluation metrics. Following Dreambooth [33], we evaluate our method with three metrics: *CLIP-I*, *CLIP-T*, and *DINO-I*. For *CLIP-I*, we calculate the CLIP visual similarity between the generated and target concept images. For *CLIP-I*, we calculate the CLIP text-image similarity between the generated images and the given text prompts. The pseudo-word (S*) in the text prompt is replaced with the proper object category for extracting CLIP text feature. For *DINO-I*, we calculate cosine similarity between the ViTS/16 DINO [5] embeddings of generated and concept images. Moreover, we adopt the *optimization time* as a metric to evaluate the efficiency of each method.

Implementation Details. We use the V1-4 version of Stable Diffusion in our experiments, and the mapping network is implemented with three-layer MLP (for both global mapping network and local mapping network). To extract multi-layer CLIP features, features from five layers are selected, and the layer indexes are $\{24, 4, 8, 12, 16\}$ in order. To train the global mapping network, we use the batch size of 16 and $\lambda_{global} = 0.01$. The learning rate is set to 1e-6. To train the local mapping network, we adopt the batch size of 8 and $\lambda_{local} = 0.0001$. The learning rate is set to 1e-5. All experiments are conducted on 4×V100 GPUs. During image generation, we use 100 steps of the LMS sampler, and the scale of classifier-free guidance is 5. Unless mentioned otherwise, we use $\lambda = 0.8$ for concept generation (*e.g*., `a photo of a S*`), and $\lambda = 0.6$ for concept editing (*e.g*., `a S* wearing sunglasses`).

4.2. Ablation Study

We first conduct the ablation studies to evaluate the effects of various components in our method, including the multi-layer features in the global mapping network, local mapping network, and the value of λ.

Effect of Multi-layer Features. Fig. 5 gives the visualization of words learned by multi-layer features in the global mapping network. For each word visualization, we use the text `A photo of a` $[w_i]$. One can see that, the word embedding of the deepest feature (*i.e*., w_0) describes the primary concept (*i.e*., corgi, teddybear), while other words describe some irrelevant details. Meanwhile, the obtained w_0 maintains superior editability. To further demonstrate this, we conducted experiments with several variants: i) Single-layer Single-word: learning a single word embedding from

Table 2. **Quantitative comparisons with existing methods.**

Method	CLIP-T (↑)	CLIP-I (↑)	DINO-I (↑)	Time (↓)
Textual Inversion [15]	0.183	0.663	0.462	50 min
DreamBooth [33]	0.251	0.785	0.674	15 min
Custom Diffusion [18]	0.245	**0.801**	**0.695**	6 min
Ours	**0.255**	0.762	0.652	**0.05s**

Table 3. **User study.** The numbers indicate the percentage (%) of volunteers who favor the results of our method over those of the competing methods based on the given question.

Metric	Ours vs. Textual Inversion	Ours vs. Dreambooth	Ours vs. Custom Diffusion
Text-alignment	75.09	57.09	62.77
Image-alignment	86.59	45.50	43.77
Editing-alignment	90.18	52.09	48.18

the deepest feature. ii) Single-layer Multi-words: learning multiple word embeddings from the deepest feature separately. iii) Multi-layers Multi-words: our setting, learning multiple word embeddings from the multiple layer features separately. Fig. 6 illustrates the results of concept generation and editing for each variant. For multiple word settings, we show the results of the full embeddings (*i.e.*, $[v]$) and the primary word embedding (denoted as $[w]$). As shown in the figure, encoding concept image into one single word embedding leads to entangled embedding with poor editability. When learning multiple words from the deepest feature, the obtained full word embeddings v and the primary word embedding w are not editable either. In contrast, the primary word w learned by our multi-features describes the object concept while maintaining superior editing capacity. That's why we only keep it during image generation. Since a single w is not sufficient to describe the details of the given concept faithfully, a local mapping network is further proposed to address this.

Effect of Local Mapping. We further conduct the ablation to evaluate the effect of the proposed local mapping network. As shown in Fig. 6, with the local mapping network, our ELITE generates images with higher consistency with the concept image. Meanwhile, from Table 1 we see that the introduction of a local mapping network does not compromise the editable capability, demonstrating its superiority over learning multiple words. Though its image alignment may not be the best, it has a good trade-off between image alignment and text alignment.

Effect of λ. In Eqn. 6, λ is introduced to control the fusion of information from the global mapping network and the local mapping network. To evaluate its effect, we vary its value from 0 to 1.2, and the generated results are shown in Fig. 7. We see that with the increase of λ, the consistency between the synthesized image and concept image is improved. However, when the value of λ is too large, it may lead to degenerated editing results. Therefore, for a good trade-off between inversion and editability, we set $\lambda = 0.6$ for editing prompts and $\lambda = 0.8$ for generating prompts. We find these parameters work well for most cases.

More ablations are provided in the *Suppl*.

4.3. Qualitative Results

To demonstrate the effectiveness of our ELITE, we compare it with existing optimization-based methods, including Textual Inversion [15], DreamBooth [33], and Custom Diffusion [18]. For a fair comparison, we trained all models using their official codes[1] and default hyperparameters on a single image. Fig. 4 illustrates the images generated with the text prompt `A photo of a S*`. With only one concept image, Textual Inversion cannot learn a word embedding to accurately describe the target concept. Although Dreambooth and Custom Diffusion learn the concept with detail consistency, their diversity may be limited. In comparison, our ELITE is capable of faithfully capturing the details of the target concept and generating diverse images. We also conduct evaluation with editing prompts and compare our method with existing methods. As shown in Fig. 8, Dreambooth and Custom Diffusion exhibit degraded editing ability, and in some cases, the editing prompts fail to produce the desired results (first row). In contrast, our method demonstrates superior editing performance. Fig. 10 illustrates more qualitative results obtained by our method. One can see that our ELITE can generate various subjects with different contexts, accessories, and properties consistently, demonstrating its effectiveness.

4.4. Quantitative Results

In addition to the qualitative comparisons, we further conduct the quantitative evaluation to validate the performance of our ELITE. From Table 2, one can see that our method achieves better text-alignment compared to the state-of-the-art methods, demonstrating its superior editability. Moreover, our method achieves comparable detail consistency and image quality, indicating its capability to generate high-quality images. Furthermore, our method provides a significant merit in terms of computational efficiency. Unlike optimization-based methods that require several or tens of minutes to obtain the concept embedding, our method can finish it in just 0.05s. This makes our method highly practical and efficient for real-world applications where speed is a critical factor.

User Study. We then perform the user study to compare with competing methods. Given a subject, a text prompt and two synthesized images (ours *v.s.* competitor), the users are asked to select the better one from three views: i) Text alignment: "Which image is more consistent with the text?". ii) Image alignment: "Which image better represents the ob-

[1]Since the official code of Dreambooth is not publicly available, we use the code implemented by https://github.com/XavierXiao/Dreambooth-Stable-Diffusion. For Textual Inversion, we use its stable diffusion version.

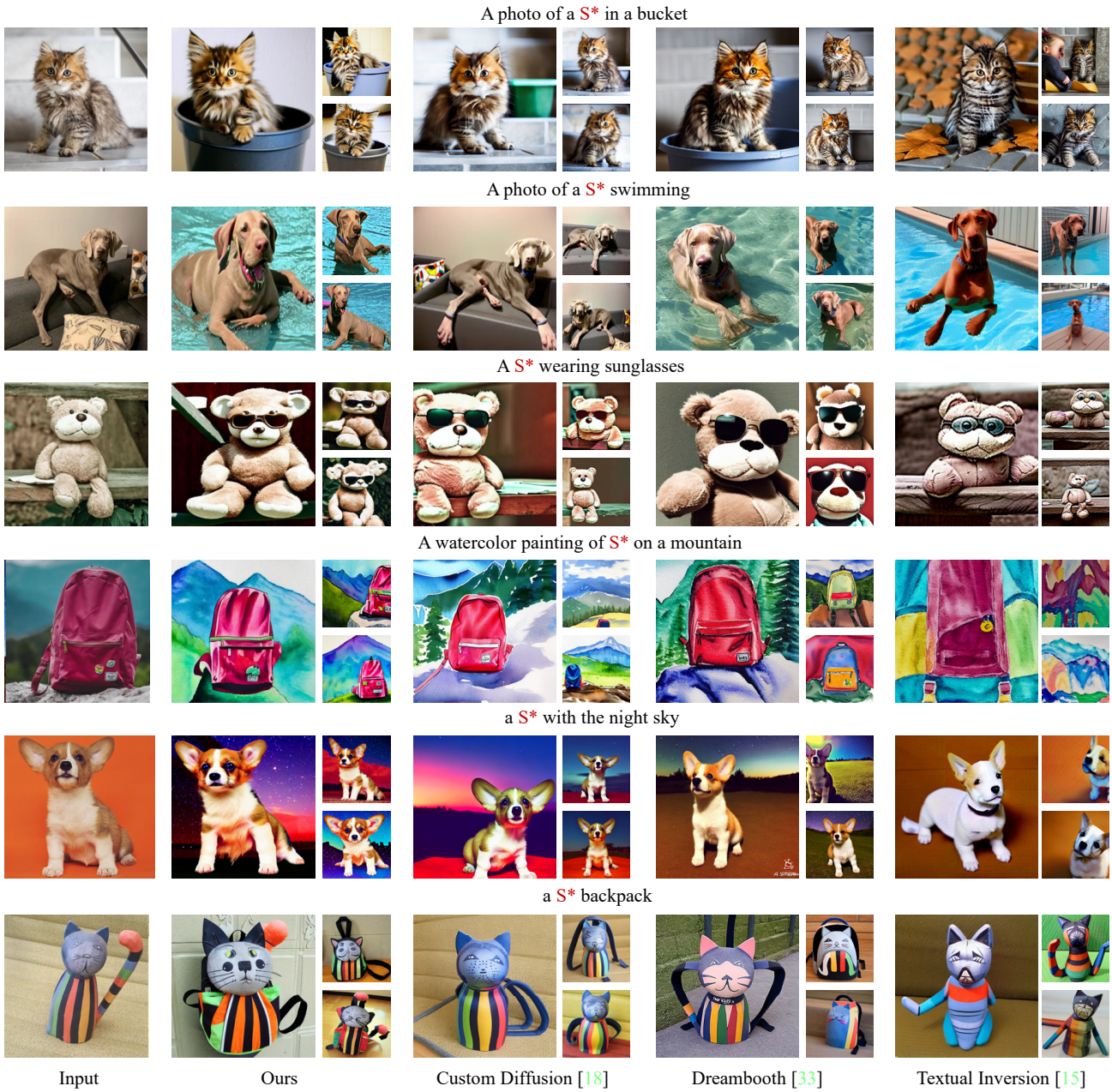


Figure 8. **Visual comparisons of concept editing.** Our ELITE method demonstrates superior editability compared to Textual Inversion [15], Dreambooth [33], and Custom Diffusion [18].

Figure 9. **Failure cases.** Similar to stable diffusion, our method fails to deal with images involving text characters.

jects in target images?". iii) Editing alignment: "Which image is more consistent with both the target subject and the text?". For each evaluated view, we employ 60 users, and each user is asked to answer 30 randomly selected questions, *i.e.*, 1800 responses in total. As shown in Table 3, our method receives comparable preference to others.

4.5. Limitations

As shown in Fig. 9, our ELITE inherits the weakness from stable diffusion, *i.e.*, failing to deal with images involving text characters.

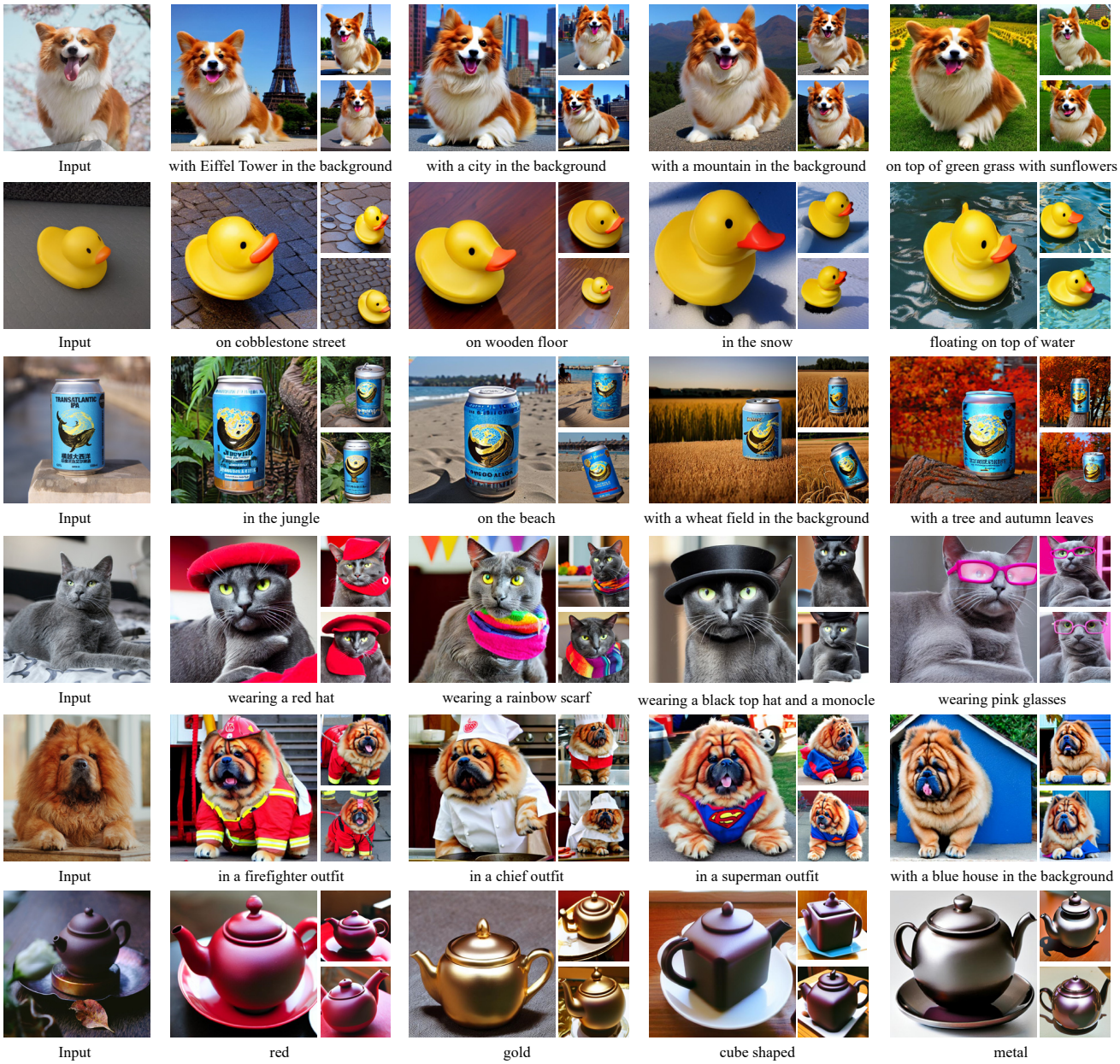


Figure 10. **More visualization results.** Our ELITE can generate subjects with different contexts, accessories and properties consistently.

5. Conclusion

In this paper, we proposed a novel learning-based encoder, namely ELITE, for fast and accurate customized text-to-image generation. Compared with existing optimization-based methods, our ELITE directly encoded visual concepts into textual embeddings, significantly reducing the computational and memory burden of learning new concepts. Moreover, our method demonstrates superior flexibility in editing the learned concepts into new scenes while preserving the image-specific details, making it a valuable tool for customized text-to-image generation. In future work, we will explore to leverage multiple concept images for better inversion, and investigate effective methods for composing multiple concepts in ELITE.

Acknowledgement. This work was supported in part by National Key R&D Program of China under Grant No. 2020AAA0104500, the National Natural Science Foundation of China (NSFC) under Grant No.s U19A2073 and 62006064, and the Hong Kong RGC RIF grant (R5001-18).

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 3

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305, 2020. 3

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 1, 2

[4] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. *arXiv preprint arXiv:1707.05776*, 2017. 2, 3

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 6

[6] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 2

[7] Guowei Chen, Yi Liu, Jian Wang, Juncai Peng, Yuying Hao, Lutao Chu, Shiyu Tang, Zewu Wu, Zeyu Chen, Zhiliang Yu, et al. Pp-matting: high-accuracy natural image matting. *arXiv preprint arXiv:2204.09433*, 2022. 5

[8] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14347–14356. IEEE, 2021. 3

[9] Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. "this is my unicorn, fluffy": Personalizing frozen vision-language representations. In *European Conference on Computer Vision*, pages 558–577. Springer, 2022. 1

[10] Antonia Creswell and Anil Anthony Bharath. Inverting the generator of a generative adversarial network. *IEEE transactions on neural networks and learning systems*, 30(7):1967–1974, 2018. 2, 3

[11] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34:19822–19835, 2021. 2

[12] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *arXiv preprint arXiv:2204.14217*, 2022. 2

[13] Ziyi Dong, Pengxu Wei, and Liang Lin. Dreamartist: Towards controllable one-shot text-to-image generation via contrastive prompt-tuning. *arXiv preprint arXiv:2211.11337*, 2022. 3

[14] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, pages 89–106. Springer, 2022. 2

[15] 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. 1, 2, 3, 4, 5, 7, 8

[16] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 1

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 5

[18] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. *arXiv preprint arXiv:2212.04488*, 2022. 1, 3, 4, 5, 7, 8

[19] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 5

[20] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. Controllable text-to-image generation. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[21] Zachary C Lipton and Subarna Tripathi. Precise recovery of latent vectors from generative adversarial networks. *arXiv preprint arXiv:1702.04782*, 2017. 2, 3

[22] Ming Liu, Yuxiang Wei, Xiaohe Wu, Wangmeng Zuo, and Lei Zhang. A survey on leveraging pre-trained generative adversarial networks for image editing and restoration. *arXiv preprint arXiv:2207.10309*, 2022. 2

[23] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022. 1, 3

[24] Minheng Ni, Zitong Huang, Kailai Feng, and Wangmeng Zuo. Imaginarynet: Learning object detectors without real images and annotations. *arXiv preprint arXiv:2210.06886*, 2022. 1

[25] 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. 2

[26] Xichen Pan, Pengda Qin, Yuhong Li, Hui Xue, and Wenhu Chen. Synthesizing coherent story with auto-regressive la-

tent diffusion models. *arXiv preprint arXiv:2211.10950*, 2022. 1

[27] Gaurav Parmar, Yijun Li, Jingwan Lu, Richard Zhang, Jun-Yan Zhu, and Krishna Kumar Singh. Spatially-adaptive multilayer selection for gan inversion and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11399–11409, 2022. 3

[28] 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. 2, 4, 5

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

[30] 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. 2

[31] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 1, 3

[32] 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. 1, 2, 3

[33] 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. *arXiv preprint arXiv:2208.12242*, 2022. 1, 3, 4, 5, 6, 7, 8

[34] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 1, 2

[35] Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. *arXiv preprint arXiv:2301.09515*, 2023. 2

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3

[37] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity gan inversion for image attribute editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11379–11388, 2022. 3

[38] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265, 2021. 2

[39] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[40] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 2

[41] Zekang Zhang, Guangyu Gao, Zhiyuan Fang, Jianbo Jiao, and Yunchao Wei. Mining unseen classes via regional objectness: A simple baseline for incremental segmentation. *Advances in Neural Information Processing Systems*, 35:24340–24353, 2022. 5