Compass Control: Multi Object Orientation Control for Text-to-Image Generation

Rishubh Parihar[1]* Vaibhav Agrawal[2]*† Sachidanand VS[1] R. Venkatesh Babu[1]
[1]IISc Bangalore [2]IIIT Hyderabad

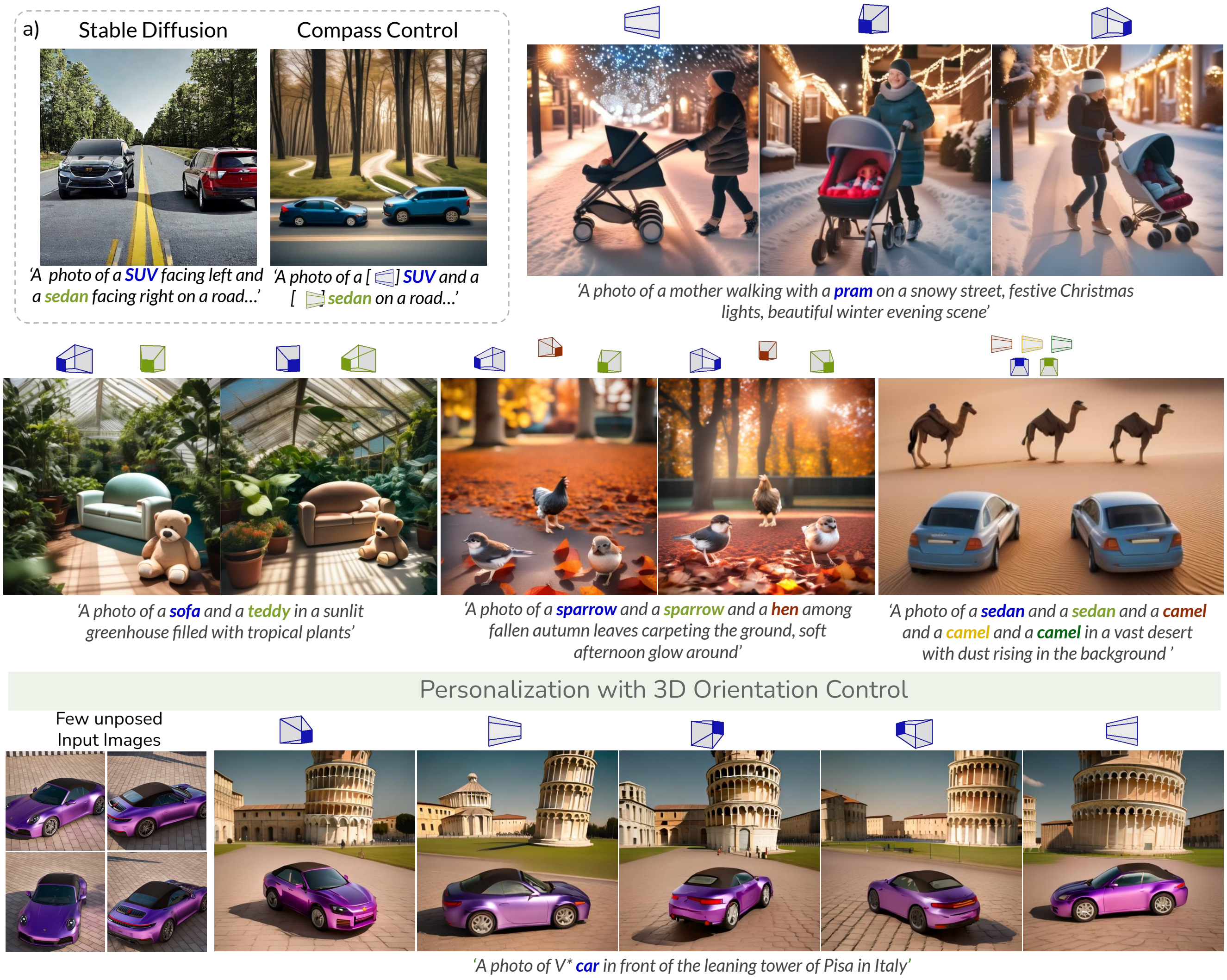

a)
Stable Diffusion
Compass Control
'A photo of a SUV facing left and a sedan facing right on a road...'
'A photo of a [ ] SUV and a [ ] sedan on a road...'
'A photo of a mother walking with a pram on a snowy street, festive Christmas lights, beautiful winter evening scene'
'A photo of a sofa and a teddy in a sunlit greenhouse filled with tropical plants'
'A photo of a sparrow and a sparrow and a hen among fallen autumn leaves carpeting the ground, soft afternoon glow around'
'A photo of a sedan and a sedan and a camel and a camel and a camel in a vast desert with dust rising in the background '
Personalization with 3D Orientation Control
Few unposed Input Images
'A photo of V* car in front of the leaning tower of Pisa in Italy'

Figure 1. We present *Compass Control*, a method to generate multi-object scenes with orientation control from text-to-image diffusion models. Given a text *prompt* and an orientation of each object (shown as frustum, the colored face is the forward direction), our method generates scenes that align with both the prompt and specified orientations. Additionally, with a few (≈ 10) unposed images of a new object, our model is personalized to generate the object in target orientations.

Abstract

Existing approaches for controlling text-to-image diffusion models, while powerful, do not allow for explicit 3D object-centric control, such as precise control of object orientation. In this work, we address the problem of multi-object orientation control in text-to-image diffusion models. This enables the generation of diverse multi-object scenes with precise orientation control for each object. The key idea is to condition the diffusion model with a set of orientation-aware ***compass*** *tokens, one for each object, along with text tokens. A light-weight encoder network predicts these compass tokens taking object orientation as the input. The model is trained on a synthetic dataset of procedurally generated scenes, each containing one or two 3D assets on a plain background. However, direct training this framework results in poor orientation control as well as leads to entanglement among objects. To mitigate this, we intervene in the generation process and constrain the cross-attention maps of each compass token to its corresponding object regions. The trained model is able to achieve precise orientation control for a) complex objects not seen during training and b) multi-object scenes with more than two objects, indicating strong generalization capabilities. Further, when combined with personalization methods, our method precisely controls the orientation of the new object in diverse contexts. Our method achieves state-of-the-art orientation control and text alignment, quantified with extensive evaluations and a user study.* project page

1. Introduction

Imagine a visual artist aiming to create a scene featuring two cars facing each other. They prompt a text-to-image model with - *'A photo of a sedan facing right and an SUV facing left'*. However, as shown in Fig. 1 (a), the generated image may not always accurately capture the intended object orientations. Moreover, relying on text prompts to control object orientation (e.g., *'facing right'*) is imprecise and requires iterative prompting. *Can we design an alternate interface for text-to-image models that accepts target orientation angle as input along with the text prompts?* Such an interface will allow for precise orientation control for each object, eliminating the need for iterative prompt adjustments and streamlining the creative process.

Several works have been proposed to achieve finer control in text-to-image models such as changing object appearance, scene layouts or image style [5, 9, 12, 22, 24, 45, 55, 57, 65, 68]. While effective for controlling 2D attributes of the image, these methods fail to accurately control 3D attributes. More recently, several methods have been proposed to control 3D properties in text-to-image models, such as camera viewpoint [7, 33], scene lighting [14], or scene layout using 3D bounding boxes [4]. However, these approaches either require dense 3D information, like multi-view images or accurate 3D bounding boxes, or limited to simple single-object scenes (Tab. 1). In this work, we present a novel interface to condition text-to-image diffusion models to generate multi-object scenes with precise 3D orientation control, without the need for multi-view images or 3D bounding boxes.

Table 1. **Comparison with Related works.** Our approach uniquely allows for object-centric orientation control and generalize to novel categories without any explicit 3D representation.

Method	CD-360 [33]	LooseControl [4]	Cont-3D-Words [14]	ViewNeTI [7]	Ours
Input	Cam Pose	3D boxes	Cam Pose	Cam Pose	Orientation
3D conditioning	Explicit	Explicit	Implicit	Implicit	Implicit
Novel classes	✗	✓	✗	✓	✓
Multiple Object Control	✓	✓	✗	✗	✓

Text-to-image (T2I) diffusion models enable the generation of objects with specific attributes via text prompts (e.g., *'a red car'*). Motivated by this, we encode the object orientation as an additional attribute in the text embedding space of the T2I model. Specifically, we introduce a special token, dubbed as ***compass*** token ($\mathbf{c}$) along with each token in the prompt (e.g., *'A photo of* $\mathbf{c_1}$ *SUV and* $\mathbf{c_2}$ *sedan on a road.'*). Each *compass* token is predicted by a lightweight encoder model taking the prescribed orientation angle as input. This formulation preserves the original interface of the base T2I model and enables precise object-centric orientation control in multi-object scenes. We train the encoder model and fine-tune the denoising U-Net with LoRA [26] on a synthetic dataset of scenes containing one or two 3D assets placed in diverse layouts on an empty floor.

We discover that directly injecting *compass* tokens with the prompt tokens leads to poor orientation control, as the *compass* token attends to irrelevant image regions, limiting its influence on its corresponding object (Fig. 4(a)). To address this, we propose **C**oupled **A**ttention **L**ocalization (**CALL**) mechanism, where we constrain the cross-attention maps of the *compass* token and its corresponding object token within a 2D bounding box. This results in a tight *coupling* between the two tokens, enabling the *compass* token to precisely control the orientation for its corresponding object. Additionally, for multi-object scenes, this results in an appropriate binding between each *compass* token and its corresponding object token, leading to disentangled orientation control of individual objects (Fig. 4(b)).

The proposed approach achieves precise orientation control for unseen objects (e.g., pram) and can generalize to scenes with more than two objects, despite being trained on one and two object scenes only (see Fig. 1). Further, given a few unposed images of a real object, we can personalize the model to control the orientation of the new object. We evaluate our method against several baselines, achieving superior performance both quantitatively and in a user study.

[*] equal contribution.

[†] work done during an internship at VAL, IISc

In summary, our primary contributions are:

1. *Compass Control* - A method for conditioning text-to-image diffusion models on object orientation, enabling precise orientation control for individual objects in multi-object scene generation.
2. *Coupled Attention Localization* - A mechanism to restrict the influence of the input object orientation to the corresponding object, ensuring effective object-centric orientation control and object disentanglement.
3. Strong generalization of *Compass Control* for precise orientation control to *unseen* objects and complex multi-object scenes, though trained on simple synthetic scenes.
4. *Personalization of Compass Control* - Given a few unposed images of a real object, our method can perform orientation control of the new object in diverse contexts.

2. Related work

Controlled generation in T2I models. Several works have been proposed to achieve fine-grained control in text-to-image diffusion models [49, 51, 53]. Recent works resort to manipulation of the text embeddings [21, 29, 44, 59, 69], or attention maps in the diffusion U-Net [1, 2, 9, 11, 16, 24, 31, 42, 45, 50, 57, 58] for controlling the generated image. Additional encoder models can be trained to condition the T2I models on a new modality such as depth, bounding boxes, or object identity [34, 65, 68]. Another set of works personalizes the diffusion model given with a few subject images [32, 52] enabling generation of the learned subject in different backgrounds. Recent work on guiding diffusion models [20, 37, 43] allows inference time control over the scene contents, allowing the control of object location, appearance, shape, and skeleton pose. However, these controls are limited to 2D.

3D-aware image editing. Recent works leverage the rich generation capability of the T2I model to perform 3D aware editing [42, 54, 61]. Specifically, they use the input scene depth as an additional input and use it to warp the internal features of the diffusion models. This enables zero-shot geometric 3D edits such as translating or rotating an object. However, these methods are limited to the editing of a single object. Another line of work uses multi-view input images and trains an implicit 3D representation such as a radiance field in the diffusion feature space [33, 46] to perform 3D consistent editing. More recently, few works leverage 3D Gaussian splat representation along with T2I models to perform scene editing [13, 38]. However, the above methods require scene-specific training and require multi-view images of accurate depth maps as input. Another direction explores large-scale training of diffusion model on a specific dataset [27, 39, 63] allowing for 3D scene editing. However, these methods fail to generalize to in-the-wild real-world scenes outside the distribution of training datasets.

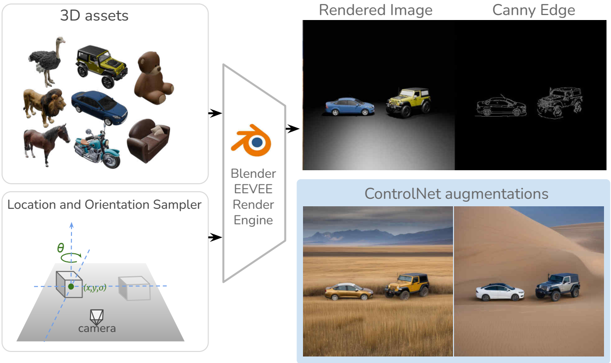


Figure 2. **Synthetic data generation.** We curate 10 diverse 3D assets, and render them in diverse layouts and orientations in Blender [15]. The rendered scenes are augmented with realistic generations from Canny [8] ControlNet [68]. The final dataset consists of one and two object scenes.

3D control in generation. Earlier works train generative models from scratch with explicit scene controls such as 3D blobs [60], or radiance fields of individual objects [3, 28, 30, 40, 41, 64] to control the generated scene. Recent works have shown the existence of 3D properties in text-to-image diffusion models [7, 17–19, 67]. This has fuelled the research that leverages this knowledge from T2I models for 3D generation. A set of works lifts this knowledge to 3D by distilling from pre-trained T2I models [35, 48, 56, 62, 66, 70]. Another set of recent works [4, 7, 14] leverage this underlying 3D knowledge for controlling 3D properties of the generated scene with an additional conditioning mechanism. ViewNeTI [7] learns a 3D view token to control the camera view for the task of novel-view synthesis from a dataset of multi-view images. ViewNeTI is natively designed for novel view synthesis but is unable to generate objects in diverse contexts. A continuous word representation is trained in [14] for 3D scene properties such as orientation and lighting on a single 3D mesh. However, both these approaches are limited to simple scenes and control only the global view angle. The closest to our work is [4] that conditions the T2I model on loose depth maps using ControlNet [68]. Loose depth maps are created using 3D object boxes and scene boundaries to enable 3D object-centric control. However, this approach relies on precise 3D boxes, making it cumbersome at inference. In contrast, our method only requires coarse 2D bounding boxes and orientation angles, offering a more user-friendly solution.

3. Method

Text-to-Image Diffusion Models. Diffusion models, when directly applied in the pixel space, are computationally expensive due to their iterative nature. To mitigate this, latent diffusion models [51] apply the diffusion process in the smaller resolution latent space of a pretrained autoencoder. Further, the generation can be conditioned on text by injecting text features into the diffusion U-Net with additional cross-attention layers. The cross-attention maps allow for precise control during generation [20, 25].

3.1. Dataset

Synthetic scene generation: We curated a list of 10 diverse 3D assets from the web: ostrich, helicopter, shoe, jeep, teddy bear, lion, sedan, horse, motorbike and sofa to aid in model generalization (see Fig. 2). Our dataset consists of 1000 one-object and 7900 two-object scenes, rendered in Blender [15]. For each image, we save the 2D bounding boxes and the orientations of the objects. The objects are placed at varied locations and in varied orientations to increase layout diversity.

Augmentations: Directly training on this dataset results in overfitting to the plain background and the black floor, as shown in ablations. To address this, we augment the dataset using ControlNet [68] to place objects in diverse backgrounds while retaining known orientations. For each rendered image, we extract its Canny [8] edge map to condition ControlNet to generate the objects in diverse contexts (for e.g., '*in a garden...*', '*near a lake...*', etc). This approach preserves object orientations while altering their appearance. We manually filter out the inconsistent augmented images. Further details on the dataset creation process can be found in the Suppl. Sec.H.

Orientation convention: In this paper, we parameterize orientation with a single angle θ, rotation around the up axis in the world coordinate system (pointing towards the sky). We define $\theta = 0$ as a reference when the object faces exactly towards the right (e.g., sedan in Fig. 2). We parameterize with a single orientation angle, as most of our objects are land objects, and only this rotation axis results in plausible object orientations. However, our method is not limited to a single orientation angle, and we present results in Suppl. Sec.B for conditioning on three orientation angles.

3.2. Compass Control

Given a text prompt $\mathcal{T}$ consisting of N object names $\{o_1, o_2, ..., o_N\}$ (e.g., '*A photo of a **jeep** and a **sedan** and a **horse** in a garden*') and their corresponding 3D orientation angles $\{\theta_1, \theta_2, ..., \theta_N\}$ we introduce a set of *compass* tokens $\{\mathbf{c_1}, \mathbf{c_2}, ..., \mathbf{c_N}\}$ to control the orientations of the respective objects. A *compass* token $\mathbf{c_n}$ is an embedding in the input space of the text encoder. It is predicted by a lightweight MLP encoder network $\mathcal{P}$ (see Fig. 3), which takes as input the object orientation angle θ_k. The compass tokens are prepended before their corresponding object tokens (e.g., '*A photo of a* $\mathbf{c_1}$ ***jeep** and a* $\mathbf{c_2}$ ***sedan** and a* $\mathbf{c_3}$ ***horse** in a garden*'), and passed through the text encoder. The outputs of the text encoder are used to condition the denoising U-Net.

However, directly training the above framework on the dataset from Sec. 3.1 fails to learn accurate orientation control. We hypothesize that this is because the added *compass* tokens are unrestricted and can attend to irrelevant image regions, limiting their influence on their corresponding objects. This is evident in the cross-attention maps for the *compass* tokens, which are indeed diffused in other image regions (see Fig. 4 (a)). Furthermore, this becomes a severe issue in multi-object scenes, where a single *compass* token can attend to multiple objects, resulting in the entanglement between different *compass* tokens(see Fig. 4 (b)). Existing works have shown that cross-attention maps closely control the image layouts [25] in the generated image. Motivated by this, we design a cross-attention localization approach.

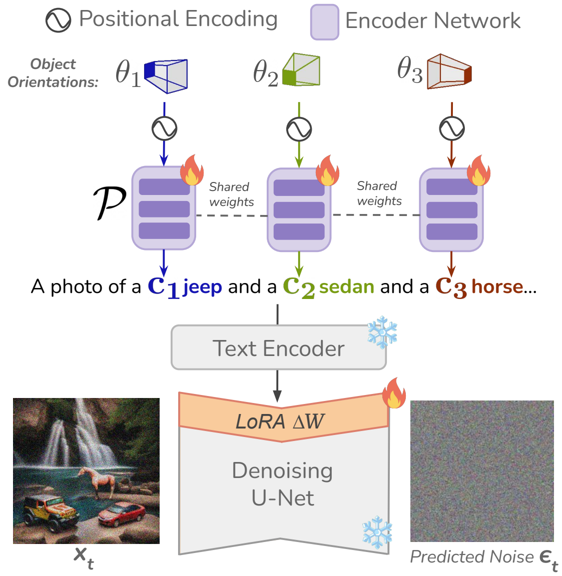


Figure 3. ***Compass Control.*** Given an orientation angle θ_j, we project it to a *compass* token with a lightweight encoder model. The *compass* tokens are interleaved with the text tokens (as shown in the figure) and passed through the text encoder. The outputs of the text encoder are used to condition the denoising process in the U-Net. We train $\mathcal{P}$ and also fine-tune the U-Net using LoRA [26].

3.3. Coupled Attention Localization (CALL)

Our key idea is to constrain the cross-attention maps for both the *compass* token and the corresponding object token inside a given 2D bounding box. This enables tight association between the object and the *compass* tokens. Additionally, it enables explicit control over the object location during generation. Specifically, during training, we use the saved object bounding boxes $\{b_1, ...b_N\}$ to compute a set of *loose square* bounding boxes $\{b^l_1, ...b^l_N\}$. The side length a_n of the loose box b^l_n is computed as $a_n = \lambda * max(h_{b_n}, w_{b_n})$, where h_{b_n} and w_{b_n} are the height and width of the object box b_n and $\lambda > 1$ is a padding factor controlling the *looseness* of the box. Next, we compute a binary mask m_n from the loose bounding box b^l_n, such that m is 0 inside the box b^l_n and $-\infty$ outside. We use it to mask out the cross attentions (Ψ) of the object token o_n and *compass* token $\mathbf{c_n}$ as follows:

$$\Psi(\mathbf{c_n}) = \text{softmax}\left(m + \frac{QK(\mathbf{c_n})^T}{\sqrt{d_K}}\right)$$

$$\Psi(\mathbf{o_n}) = \text{softmax}\left(m + \frac{QK(o_n)^T}{\sqrt{d_K}}\right)$$

where the query feature Q comes from the U-Net features, and the key features $K(\mathbf{c_n})$ and $K(\mathbf{o_n})$ come from the respective tokens. This masking operation is performed at all the diffusion timesteps and cross-attention layers. We find that using a loose mask is highly effective, providing greater flexibility during the generation process. This attention localization mechanism is dubbed as *Coupled Attention Localization*, or *CALL* for short. Adding CALL mechanism during training and inference has key advantages for composing multi-object scenes: *a)* Appropriate binding of each compass token $\mathbf{c_n}$ with its corresponding object token o_n leads to disentangled orientation control of individual objects. *b)* Constraining the cross-attention for the object tokens o_n to *non-overlapping* bounding boxes results in disentanglement between the objects themselves (a known issue in T2I models [11]), *enabling strong generalization to complex scenes with multiple objects*.

Training. We train our encoder model $\mathcal{P}$ and fine-tune the denoising U-Net with LoRA [26] on the synthetic dataset from Sec. 3.1. The LoRA training is extremely parameter efficient and preserves the behavior of the base T2I model. We use the proposed CALL mechanism for effective learning of orientation control. However, for the effective working of CALL, the object must be generated within the loose bounding box, as the *compass* token's influence is restricted to this region. To this end, we first train on simple single-object scenes, to first learn the bounding box adherence for the generated object, and then train on a mix of both single and two-object scenes thereafter. We contrast this two-staged training procedure with the single-stage training at an intermediate training iteration in Fig. 5. The generated objects adhere to the bounding box better in the two-staged training compared to single-stage training This leads to effective learning of orientation control, as we have shown in ablative experiments (Sec. 4).

Inference. During the inference phase, *Compass Control* expects the text prompt containing the objects, desired orientations, and optional coarse 2D bounding boxes as input. *Using loose bounding boxes during training offers a significant advantage here, as we can even spawn non-overlapping boxes heuristically*, as shown in the Suppl. Sec.F. We use the text tokens and the *compass* tokens together to condition the diffusion model.

3.4. Personalization

The design of *Compass Control* as a conditioning mechanism preserves the original capabilities of T2I models, such as personalization. Given a few unposed images of an object (≈ 10), we apply Dreambooth [52] with LoRA and associate a special token $\hat{u}$ for the input object using *Compass Control*'s fine-tuned UNet. During inference, we can generate the object in desired orientation θ with simple prompts; e.g., *'A photo of a* $\mathbf{c}(\theta)$ $\hat{u}$ *car on the beach.'*, where $\mathbf{c}(\theta)$ is the *compass* token.

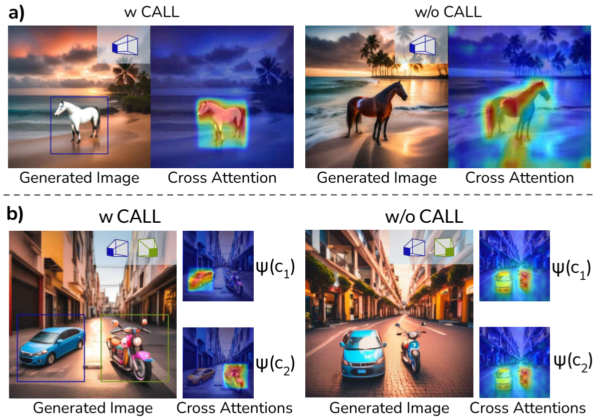


Figure 4. **Binding the *compass* tokens:** We visualize the averaged cross attention of the *compass* token(s) when training with CALL (shown on the left) and without it (shown on the right). CALL localizes the influence of the *compass* token at the *correct* regions, which (a) improves orientation control (b) disentangles orientations in multi-object scenes. In (b), $\mathbf{c_1}$ and $\mathbf{c_2}$ are compass tokens for car and motorbike, respectively.

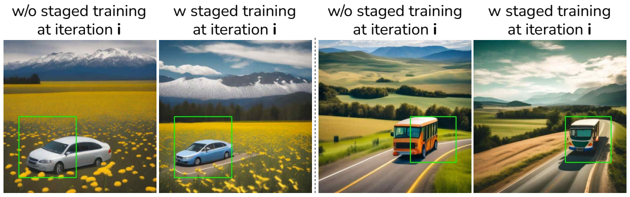

w/o staged training at iteration **i**
w staged training at iteration **i**
w/o staged training at iteration **i**
w staged training at iteration **i**

Figure 5. **Staged training** results in improved adherence of objects to the bounding boxes, leading to orientation learning.

4. Experiments

4.1. Experiment setup

Dataset. We use the synthetic dataset from Sec. 3.1 consisting of 8900 rendered images and 6010 augmented images consisting of one and two object scenes for training our model. For quantitative evaluation, we construct a test set of 10 scene prompts having one/two object names generated from ChatGPT [10]. We use a mix of seen objects (*horse, jeep, sedan, sofa, teddy, lion*) and unseen objects (*boat, dolphin, ship, SUV, tractor*) in the prompts. For each object and prompt combination, we use a set of 10 randomly sampled orientations. The list of input text prompts and orientations is given in the Supp. Sec.L.

Implementation Details. We use Stable Diffusion

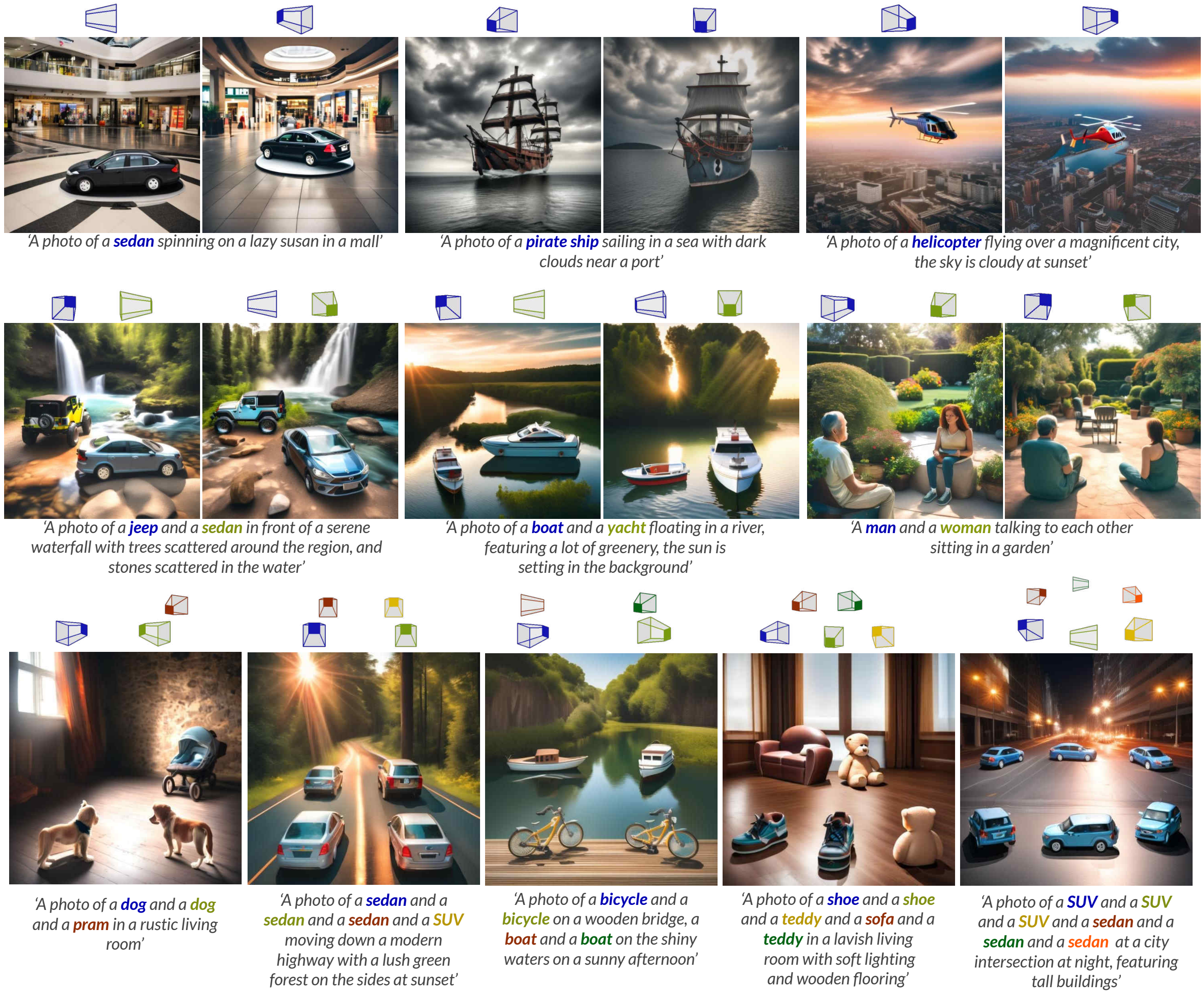

'A photo of a sedan spinning on a lazy susan in a mall'
'A photo of a pirate ship sailing in a sea with dark clouds near a port'
'A photo of a helicopter flying over a magnificent city, the sky is cloudy at sunset'
'A photo of a jeep and a sedan in front of a serene waterfall with trees scattered around the region, and stones scattered in the water'
'A photo of a boat and a yacht floating in a river, featuring a lot of greenery, the sun is setting in the background'
'A man and a woman talking to each other sitting in a garden'
'A photo of a dog and a dog and a pram in a rustic living room'
'A photo of a sedan and a sedan and a sedan and a SUV moving down a modern highway with a lush green forest on the sides at sunset'
'A photo of a bicycle and a bicycle on a wooden bridge, a boat and a boat on the shiny waters on a sunny afternoon'
'A photo of a shoe and a shoe and a teddy and a sofa and a teddy in a lavish living room with soft lighting and wooden flooring'
'A photo of a SUV and a SUV and a SUV and a sedan and a sedan and a sedan at a city intersection at night, featuring tall buildings'

Figure 6. **Main Results.** *Compass Control* generates complex scenes aligned with the text prompts and the orientations (shown as frustum, the colored face is forward direction). It generalizes well to unseen object categories - *pirate ship, boat, yacht, bicycle, pram, dog, and even human*. Further, it generates high-quality compositions of several objects, despite being *trained only on one and two object scenes.*

v2.1 [51] as our base T2I model and use LoRA rank 4 for fine-tuning its UNet. We present additional results on Stable Diffusion-XL [47] in Suppl. Sec. D. Our encoder model $\mathcal{P}$ is a lightweight MLP: three linear layers with ReLU. We train our model for $25,000$ steps with a batch size of 4 with AdamW optimizer and a fixed learning rate of 10^{-4}. We keep the bounding box padding $\lambda = 1.2$ for CALL. The training takes 24 hours on a single A6000 GPU.

Evaluation Metrics. We evaluate for a) *Text Alignment* using CLIP similarly; b) *%Object Generation* - we evaluate the presence of the intended object using Grounding-DINO [36] and threshold on the objectness score for each object in the prompt. c) *Angular Error* - to evaluate the orientation consistency, we compute the Angular error (in radians) between the input orientation angle θ and the orientation of the generated object using a pretrained orientation predictor. Further details about the orientation predictor and the implementation details of the metrics are in the Suppl Sec.I.

Baselines. As no prior method tackles our same task, we compare against methods that allow for either camera pose control or 3D object pose control in text-to-image models: a) Continuous 3D Words (*Cont-3D-Words*) [14]: Following their exact setup, we train a 3D word for controlling the object orientation on renderings of a single 3D asset - Sedan and its ControlNet augmentations. b) *ViewNeTI* [7]: In contrast to Cont-3D-Words, ViewNeTI allows for training on multiple 3D assets; for fair evaluation, we train ViewNeTI on our training dataset and condition the T2I on object orientation instead of 3D camera pose. Notably, both of these methods are limited to a global view control. Hence, we evaluate on only single-object scenes. c) *LooseControl* [4], allows for multi-object control by conditioning on loose depth maps (Fig. 7) formed by 3D object boxes. We use template 3D bounding boxes for each test object and place them in the scene, with random orientation and location (similar to Sec. 3.1). For a fair comparison, we use the 2D boxes corresponding to the 3D boxes in our outputs. Notably, LooseControl does not take exact orientation as input as a 180 flipped box also has the same *loose* depth; we consider this while computing the Angular error. Further, it requires the user to provide *accurate 3D bounding boxes during inference,* which is cumbersome, whereas our method requires only loose 2D boxes. Additional baseline details are in Suppl. Sec.J.

4.2. Main Results

Qualitative results. We present our method's results in Fig. 6. Our method is able to generate complex multi-object scenes with precise orientation control of individual objects, even though it is *trained with single and two object scenes.* Further, it is able to generalize well to challenging unseen objects such as *humans* and *prams*. Interestingly, there was no water-based subject in the training dataset, yet our method can achieve precise orientation control for a *ship*, *yacht*, and *boat*. These strong generalization capabilities of *Compass Control* can be attributed to effective attention constraining with CALL and diversity in the 3D assets used for training. The conditioning mechanism of Compass Control is generalized, and we present results for jointly controlling all *three orientation angles*, *camera elevation*, and *object scale* in Suppl. Sec.B & C.

Baseline comparison. We compare our method to all three baselines on single-object scenes and additionally include multi-object scene comparison with LooseControl in Fig. 7 and Tab. 2. Cont-3D-words morphs the generated objects into a *sedan* shape seen during training and generates washed-out backgrounds. This results in poor text alignment and a lower percentage of intended object generation. ViewNeTI can generate better object shapes in a given orientation; however, it overfits to the black backgrounds seen during training, leading to poor text alignment. This is primarily because ViewNeTI does not accept ControlNet augmentations in its original form as it is designed for novel view synthesis. LooseControl generates realistic single-subject scenes following the given text prompt. However, in some cases, the object orientation is not followed (e.g., *sofa, teddy*). For a multi-object generation (Fig. 7b)), LooseControl either misses the object during generation (*horse* in the second column and cars in the last column) or distorts the object shapes. LooseControl distorts the object to a *box* like appearance (columns 3 and 4) as it learns LoRA over the original depth-conditioned ControlNet, which follows the depth input closely, resulting in an inferior object generation score. Further, though we adjust for a 180-degree flip in computing Angular error for LooseControl, our method achieves significantly lower Angular error, demonstrating a strong orientation adherence.

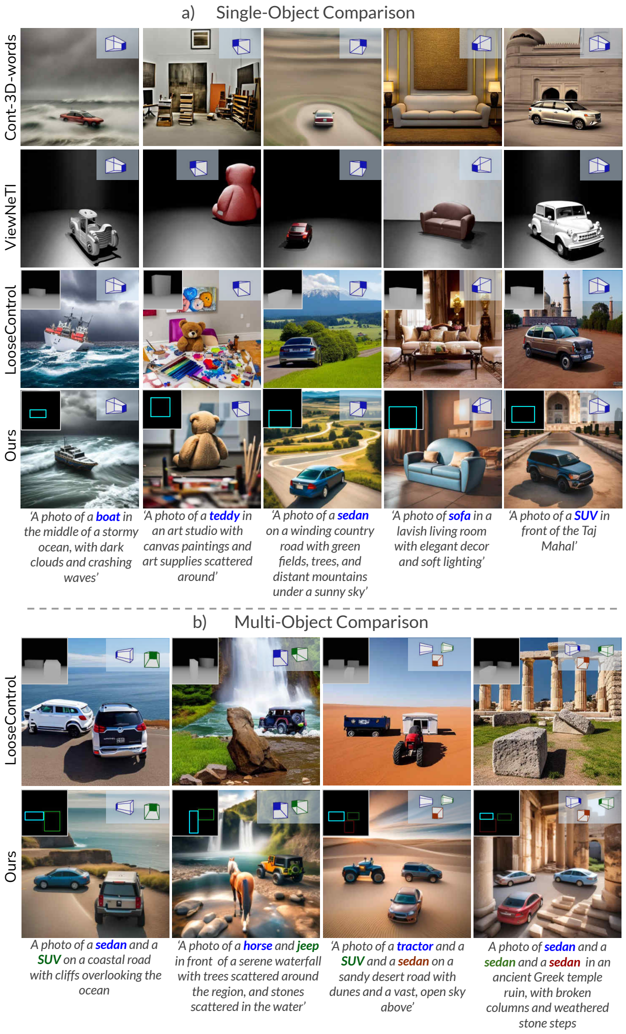

a) Single-Object Comparison
Cont-3D-words
ViewNeTI
LooseControl
Ours
'A photo of a boat in the middle of a stormy ocean, with dark clouds and crashing waves'
'A photo of a teddy in an art studio with canvas paintings and art supplies scattered around'
'A photo of a sedan on a winding country road with green fields, trees, and distant mountains under a sunny sky'
'A photo of sofa in a lavish living room with elegant decor and soft lighting'
'A photo of a SUV in front of the Taj Mahal'
b) Multi-Object Comparison
LooseControl
Ours
A photo of a sedan and a SUV on a coastal road with cliffs overlooking the ocean
'A photo of a horse and jeep in front of a serene waterfall with trees scattered around the region, and stones scattered in the water'
'A photo of a tractor and a SUV and a sedan on a sandy desert road with dunes and a vast, open sky above'
A photo of sedan and a sedan and a sedan in an ancient Greek temple ruin, with broken columns and weathered stone steps

Figure 7. **Qualitative Comparison.** We compare our method against three baselines. Cont-3D-words [14] does not generate the intended object whereas View-NeTI [7] generates objects in plain backgrounds. LooseControl [4] generates realistic compositions but does not follow the input orientation well. In contrast, our method aligns with the input text prompt and follows the input orientation, while generating realistic scenes.

4.3. User study

We conducted a user study with 57 participants to compare all methods on *text alignment*, *object quality*, and *orientation consistency*. Users rated 90 image pairs for single-

Single object	Text Align. ↓	% Obj. Generated ↑	Angular Err. ↓
ViewNeTI [7]	22.12	0.920	0.596
Cont-3D-words [14]	29.88	0.732	0.509
LooseControl [4]	31.60	0.656	0.385
Ours	**32.98**	**0.968**	**0.198**
Multiple object	**Text Align. ↓**	**% Obj. Generated ↑**	**Angular Err. ↓**
LooseControl [4]	31.73	0.778	0.372
Ours	**33.93**	**0.964**	**0.215**

Table 2. **Quantiative comparison.** We compute Text Alignment, using CLIP, % of correct subject generation and Angular error between the predicted and input orientations.

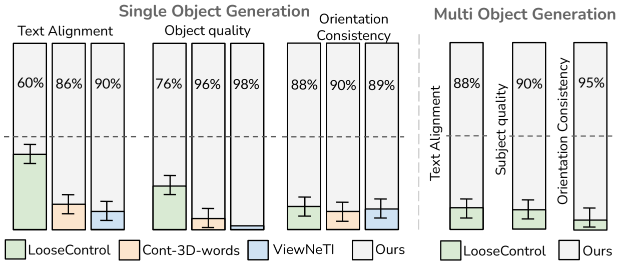


Figure 8. **User study.** We compare all methods on the three image metrics. Each bar indicates the fraction of people that preferred our result (gray) vs the baseline (other color).

object scenes and 30 for multi-object scenes, choosing the better image in each pair, sampled from our method and baselines. In total, we obtained 5130 ratings for single-object and 1710 for two-object scenes. Results (Fig. 8) show users preferred our method overall, with LooseControl scoring well in text alignment for single objects but falling short across all metrics in multi-object scenes.

4.4. Personalization

We present personalization results in Fig. 9. With only 10 unposed images of an object, our method can generate the object with precise orientation control in various contexts. Furthermore, we can jointly optimize Dreambooth [52] LoRA weights for two objects (e.g., a *chair* and a *teddy bear*), enabling multi-object personalization with object-centric orientation control. We compare our method with CD-360 [33] and achieve comparable performance. Unlike CD-360 [33], which requires ≈ 100 object images with camera pose, we require only a few unposed images, making it more convenient and user-friendly.

4.5. Ablations

We present the results of the ablation study in Fig. 10 on generated scenes with 1, 3, and 5 objects. We focus on three key design choices and generate scenes with a variable number of objects:

Staged training: Training *Compass Control* in a single stage results in poor adherence to the bounding boxes. This especially affects complex multi-object layouts, where some objects tend to *leak* outside their box and suppress the generation of neighboring objects. This results in generating a lesser number of objects in the scene.

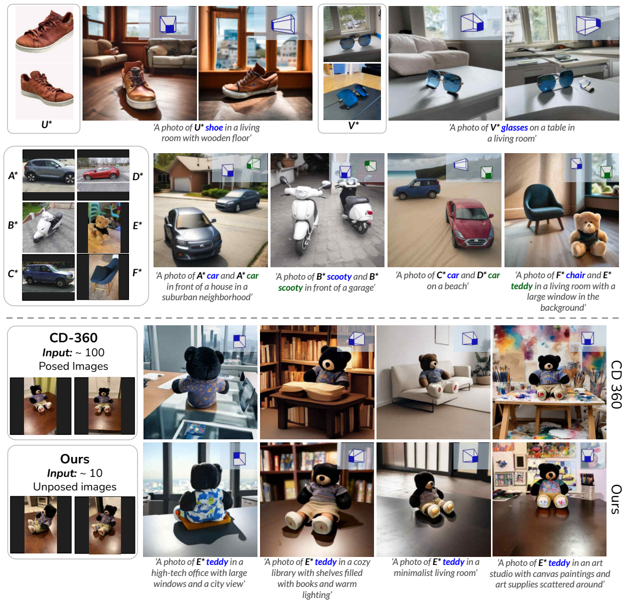

U*
'A photo of U* shoe in a living room with wooden floor'
V*
'A photo of V* glasses on a table in a living room'
A*
B*
C*
D*
E*
F*
'A photo of A* car and A* car in front of a house in a suburban neighborhood'
'A photo of B* scooty and B* scooty in front of a garage'
'A photo of C* car and D* car on a beach'
'A photo of F* chair and E* teddy in a living room with a large window in the background'
CD-360
Input: ~ 100
Posed Images
CD 360
Ours
Input: ~ 10
Unposed images
Ours
'A photo of E* teddy in a high-tech office with large windows and a city view'
'A photo of E* teddy in a cozy library with shelves filled with books and warm lighting'
'A photo of E* teddy in a minimalist living room'
'A photo of E* teddy in an art studio with canvas paintings and art supplies scattered around'

Figure 9. **Personalization.** Given a few (≈ 10) unposed images of an object, our method can personalize the diffusion models and allow for orientation control of the new object. Notably, our method can also generate scenes with two personalized objects with precise orientation control. Additionally, we compare our method with CustomDiffusion-360 [33] that uses ≈ 100 posed images.

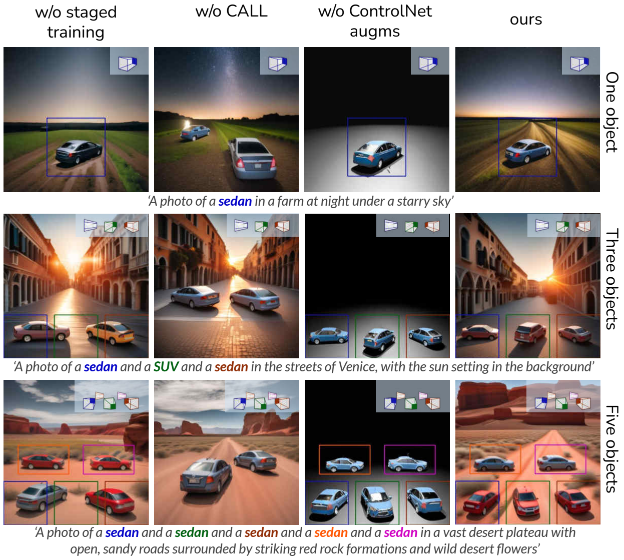

w/o staged training
w/o CALL
w/o ControlNet augms
ours
One object
'A photo of a sedan in a farm at night under a starry sky'
Three objects
'A photo of a sedan and a SUV and a sedan in the streets of Venice, with the sun setting in the background'
Five objects
'A photo of a sedan and a sedan and a sedan and a sedan and a sedan in a vast desert plateau with open, sandy roads surrounded by striking red rock formations and wild desert flowers'

Figure 10. **Ablation studies.** We show the impact of several design choices of our approach. Refer Sec. 4.5 for details.

CALL is crucial for accurate orientation control. Without it the *compass* token attends to irrelevant image regions, resulting in poor orientation control. Further, without CALL the object tokens entangle with each other during generation (a known issue in T2I models [11]). This results in generating a lesser number of objects in the scene.

Augmentations: Without the ControlNet augmentations, the model overfits the training backgrounds, resulting in black backgrounds. Hence, ControlNet augmentation is necessary to generate objects in diverse contexts.

5. Conclusion and Discussion

Limitations. Our method struggles to control the orientation of objects that are occluded or have significant overlap. In these scenarios, the model either fails to generate one of the objects or mixes the attributes between objects (Fig. 11). Further, our single-angle orientation representation is too simplistic for modeling complex non-rigid objects such as humans. In this work, we focus on presenting a generalized framework to condition text-to-image models on 3D controls that can be adapted easily for other representations.

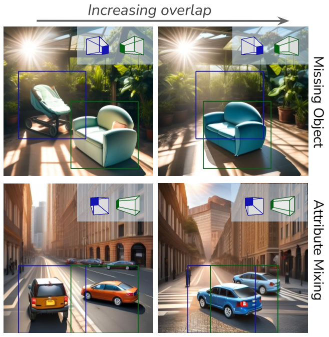


Figure 11. Failure Cases

Conclusion. In this work, we propose a method to condition pre-trained text-to-image models with 3D object orientation while preserving its rich image-generation capabilities. We train the conditioning module on a small synthetic dataset via staged training and involving attention regularization. These modifications enable strong generalization of the model, allowing for precise orientation control for unseen object categories and individual objects in a multi-object scene. Further, it can be seamlessly integrated with personalization methods to achieve orientation control of personalized objects. This work is a testament that text-to-image diffusion models innately have some form of 3D understanding, and interesting 3D controls can be obtained with appropriate conditionings.

Acknowledgements. We thank Srinjay Sarkar, Abhijnya Bhat, and Tejan Karmali for thoroughly reviewing the manuscript. This work is supported by Meesho and PMRF by the Government of India.

References

[1] Omri Avrahami, Rinon Gal, Gal Chechik, Ohad Fried, Dani Lischinski, Arash Vahdat, and Weili Nie. Diffuhaul: A training-free method for object dragging in images. *arXiv preprint arXiv:2406.01594*, 2024. 3

[2] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. 2023. 3

[3] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, et al. Gaudi: A neural architect for immersive 3d scene generation. *Advances in Neural Information Processing Systems*, 35:25102–25116, 2022. 3

[4] Shariq Farooq Bhat, Niloy Mitra, and Peter Wonka. Loosecontrol: Lifting controlnet for generalized depth conditioning. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2, 3, 7, 8, 1, 4, 6

[5] Manuel Brack, Felix Friedrich, Katharia Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinário Passos. Ledits++: Limitless image editing using text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8861–8870, 2024. 2

[6] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 4

[7] James Burgess, Kuan-Chieh Wang, and Serena Yeung. Viewpoint textual inversion: Unleashing novel view synthesis with pretrained 2d diffusion models. *arXiv preprint arXiv:2309.07986*, 2023. 2, 3, 6, 7, 8, 1

[8] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 3, 4

[9] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023. 2, 3

[10] Chatgpt. Chatgpt. In *https://chat.openai.com/chat*, 2022. 5

[11] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023. 3, 5, 9

[12] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6593–6602, 2024. 2

[13] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. corr abs/2311.14521 (2023), 2023. 3

[14] Ta-Ying Cheng, Matheus Gadelha, Thibault Groueix, Matthew Fisher, Radomir Mech, Andrew Markham, and Niki Trigoni. Learning continuous 3d words for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6753–6762, 2024. 2, 3, 6, 7, 8, 1

[15] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. 3, 4, 8

[16] Omer Dahary, Or Patashnik, Kfir Aberman, and Daniel Cohen-Or. Be yourself: Bounded attention for multi-subject

text-to-image generation. *arXiv preprint arXiv:2403.16990*, 2(5), 2024. 3

[17] Ankit Dhiman, Manan Shah, Rishubh Parihar, Yash Bhalgat, Lokesh R Boregowda, and R Venkatesh Babu. Reflecting reality: Enabling diffusion models to produce faithful mirror reflections. *arXiv preprint arXiv:2409.14677*, 2024. 3

[18] Xiaodan Du, Nicholas Kolkin, Greg Shakhnarovich, and Anand Bhattad. Generative models: What do they know? do they know things? let's find out! *arXiv preprint arXiv:2311.17137*, 2023.

[19] Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21795–21806, 2024. 3

[20] Dave Epstein, Allan Jabri, Ben Poole, Alexei Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *Advances in Neural Information Processing Systems*, 36:16222–16239, 2023. 3, 4

[21] Songwei Ge, Taesung Park, Jun-Yan Zhu, and Jia-Bin Huang. Expressive text-to-image generation with rich text. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7545–7556, 2023. 3

[22] Daniel Geng and Andrew Owens. Motion guidance: Diffusion-based image editing with differentiable motion estimators. In *The Twelfth International Conference on Learning Representations*. 2

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[24] 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. 2, 3

[25] 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. 4

[26] 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. 2, 4, 5, 7

[27] Allan Jabri, Sjoerd van Steenkiste, Emiel Hoogeboom, Mehdi SM Sajjadi, and Thomas Kipf. Dorsal: Diffusion for object-centric representations of scenes et. al. *arXiv preprint arXiv:2306.08068*, 2023. 3

[28] Kunal Kathare, Ankit Dhiman, K Vikas Gowda, Siddharth Aravindan, Shubham Monga, Basavaraja Shanthappa Vandrotti, and Lokesh R Boregowda. Instructive3d: Editing large reconstruction models with text instructions. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pages 3246–3256, 2025. 3

[29] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 3

[30] Hyunsu Kim, Gayoung Lee, Yunjey Choi, Jin-Hwa Kim, and Jun-Yan Zhu. 3d-aware blending with generative nerfs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22906–22918, 2023. 3

[31] Yunji Kim, Jiyoung Lee, Jin-Hwa Kim, Jung-Woo Ha, and Jun-Yan Zhu. Dense text-to-image generation with attention modulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7701–7711, 2023. 3

[32] 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. 3

[33] Nupur Kumari, Grace Su, Richard Zhang, Taesung Park, Eli Shechtman, and Jun-Yan Zhu. Customizing text-to-image diffusion with camera viewpoint control. *arXiv preprint arXiv:2404.12333*, 2024. 2, 3, 8

[34] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 3

[35] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[36] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 6, 5

[37] Grace Luo, Trevor Darrell, Oliver Wang, Dan B Goldman, and Aleksander Holynski. Readout guidance: Learning control from diffusion features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8217–8227, 2024. 3

[38] Guan Luo, Tian-Xing Xu, Ying-Tian Liu, Xiao-Xiong Fan, Fang-Lue Zhang, and Song-Hai Zhang. 3d gaussian editing with a single image. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6627–6636, 2024. 3

[39] Oscar Michel, Anand Bhattad, Eli VanderBilt, Ranjay Krishna, Aniruddha Kembhavi, and Tanmay Gupta. Object 3dit: Language-guided 3d-aware image editing. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[40] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7588–7597, 2019. 3

[41] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer*

Vision and Pattern Recognition, pages 11453–11464, 2021. 3

[42] Karran Pandey, Paul Guerrero, Matheus Gadelha, Yannick Hold-Geoffroy, Karan Singh, and Niloy J Mitra. Diffusion handles enabling 3d edits for diffusion models by lifting activations to 3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7695–7704, 2024. 3

[43] Rishubh Parihar, Abhijnya Bhat, Abhipsa Basu, Saswat Mallick, Jogendra Nath Kundu, and R Venkatesh Babu. Balancing act: distribution-guided debiasing in diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6668–6678, 2024. 3

[44] Rishubh Parihar, VS Sachidanand, Sabariswaran Mani, Tejan Karmali, and R Venkatesh Babu. Precisecontrol: Enhancing text-to-image diffusion models with fine-grained attribute control. In *European Conference on Computer Vision*, pages 469–487. Springer, 2024. 3

[45] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23051–23061, 2023. 2, 3

[46] Or Patashnik, Rinon Gal, Daniel Cohen-Or, Jun-Yan Zhu, and Fernando De la Torre. Consolidating attention features for multi-view image editing. *arXiv preprint arXiv:2402.14792*, 2024. 3

[47] 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, 2023. 6

[48] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 3

[49] 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. 3

[50] Harsh Rangwani, Aishwarya Agarwal, Kuldeep Kulkarni, R Venkatesh Babu, and Srikrishna Karanam. Crafting parts for expressive object composition. *arXiv preprint arXiv:2406.10197*, 2024. 3

[51] 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. 3, 6, 5, 7, 8

[52] 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. 3, 5, 8

[53] 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. 3

[54] Rahul Sajnani, Jeroen Vanbaar, Jie Min, Kapil Katyal, and Srinath Sridhar. Geodiffuser: Geometry-based image editing with diffusion models. *arXiv preprint arXiv:2404.14403*, 2024. 3

[55] Yizhi Song, Zhifei Zhang, Zhe Lin, Scott Cohen, Brian Price, Jianming Zhang, Soo Ye Kim, and Daniel Aliaga. Objectstitch: Object compositing with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18310–18319, 2023. 2

[56] Alexander Vilesov, Pradyumna Chari, and Achuta Kadambi. Cg3d: Compositional generation for text-to-3d via gaussian splatting. *arXiv preprint arXiv:2311.17907*, 2023. 3

[57] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 2, 3

[58] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3

[59] 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. 3

[60] Qian Wang, Yiqun Wang, Michael Birsak, and Peter Wonka. Blobgan-3d: A spatially-disentangled 3d-aware generative model for indoor scenes. *arXiv preprint arXiv:2303.14706*, 2023. 3

[61] Ruicheng Wang, Jianfeng Xiang, Jiaolong Yang, and Xin Tong. Diffusion models are geometry critics: Single image 3d editing using pre-trained diffusion priors. *arXiv preprint arXiv:2403.11503*, 2024. 3

[62] Zhaoning Wang, Ming Li, and Chen Chen. Luciddreaming: Controllable object-centric 3d generation. *arXiv preprint arXiv:2312.00588*, 2023. 3

[63] Ziyi Wu, Yulia Rubanova, Rishabh Kabra, Drew A Hudson, Igor Gilitschenski, Yusuf Aytar, Sjoerd van Steenkiste, Kelsey R Allen, and Thomas Kipf. Neural assets: 3d-aware multi-object scene synthesis with image diffusion models. *arXiv preprint arXiv:2406.09292*, 2024. 3, 4, 6

[64] Yang Xue, Yuheng Li, Krishna Kumar Singh, and Yong Jae Lee. Giraffe hd: A high-resolution 3d-aware generative model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18440–18449, 2022. 3

[65] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 2, 3

[66] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. *arXiv preprint arXiv:2406.09394*, 2024. 3

[67] Guanqi Zhan, Chuanxia Zheng, Weidi Xie, and Andrew Zisserman. What does stable diffusion know about the 3d scene? *arXiv preprint arXiv:2310.06836*, 2023. 3

[68] 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. 2, 3, 4

[69] Yuxin Zhang, Weiming Dong, Fan Tang, Nisha Huang, Haibin Huang, Chongyang Ma, Tong-Yee Lee, Oliver Deussen, and Changsheng Xu. Prospect: Prompt spectrum for attribute-aware personalization of diffusion models. *ACM Transactions on Graphics (TOG)*, 42(6):1–14, 2023. 3

[70] Shijie Zhou, Zhiwen Fan, Dejia Xu, Haoran Chang, Pradyumna Chari, Tejas Bharadwaj, Suya You, Zhangyang Wang, and Achuta Kadambi. Dreamscene360: Unconstrained text-to-3d scene generation with panoramic gaussian splatting. In *European Conference on Computer Vision*, pages 324–342. Springer, 2025. 3

Compass Control: Multi Object Orientation Control for Text-to-Image Generation

Supplementary Material

Contents

A. Project page

Check the `project page` for interactive visualizations of 3D orientation control.

B. Controlling 3D orientation

The main text primarily focused on an orientation control for a single angle. However, our method is not limited to single orientation control, and we present an experiment for controlling all three orientation angles in a single model. Specifically, we updated the pose injection network to take 3 orientation angles as input to predict the pose token. We trained the model on flying objects - airplanes and helicopters- as rotation along all three axes is plausible for these objects. Specifically, we used six 3D assets from the web for these categories and followed the procedure in sec. 3.1 (main paper) to render the dataset. We present results for controlling all the 3 orientation angles in Fig. 13 and 14. In Fig. 13, we present rotation along all three axes for a fighter jet aircraft in three separate rows. Observe that, our method can precisely control all the object orientations along all the three axis. In Fig. 14, we show the generalization of our trained model in controlling the orientation of a variety of objects. Notably, our model is not trained on birds or rockets. Still, it can generate consistent orientation-conditioned scenes following the text prompts. Note, that the compass shown in the figure is just for visualization purposes (can have an error of a few degrees).

C. Additional Control

Continuous control for camera elevation. Our proposed conditioning mechanism is generalized and can be adapted to achieve continuous camera elevation control in Fig. 12. We generated a dataset with camera elevation variations and conditioned the denoising UNet on elevation angle.

Control for object scale. We can also precisely control the size of individual objects with additional conditioning on the object scale, as shown in Fig. 12. Specifically, we condition the diffusion model with the length of the diagonal of a tight 2D bounding box.

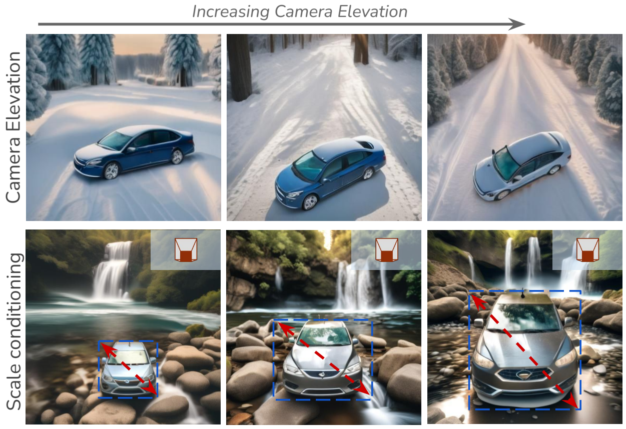


Figure 12. Additional Controls: (Top) Conditioning with camera elevation angle. (Bottom) Conditioning on object scale.

D. Generalization to StableDiffusion-XL

We have presented all the results on StableDiffusion-2.1 in the main paper. Our method also generalizes well to a larger

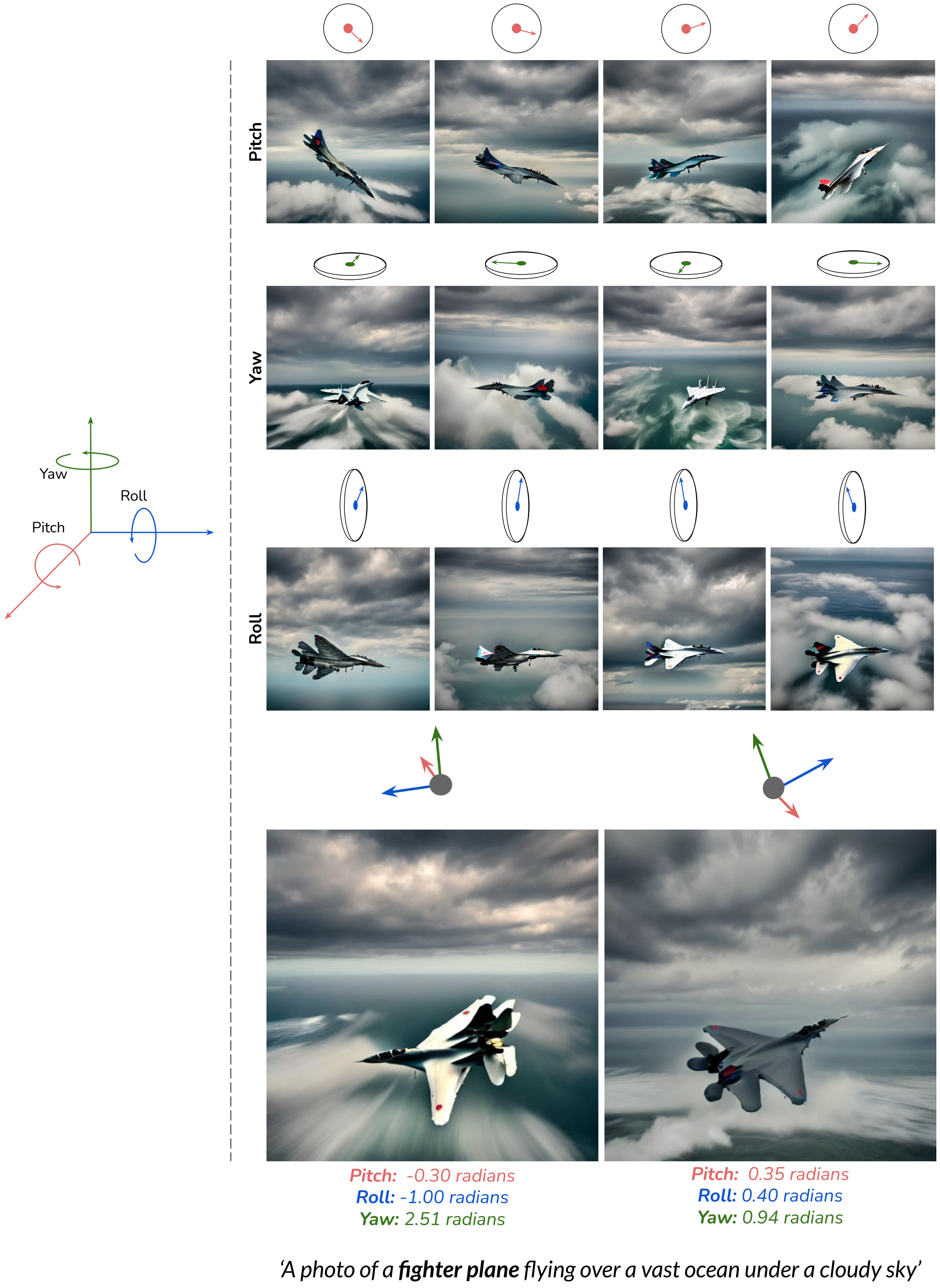


Figure 13. Conditioning on all three orientation angles for a single object.

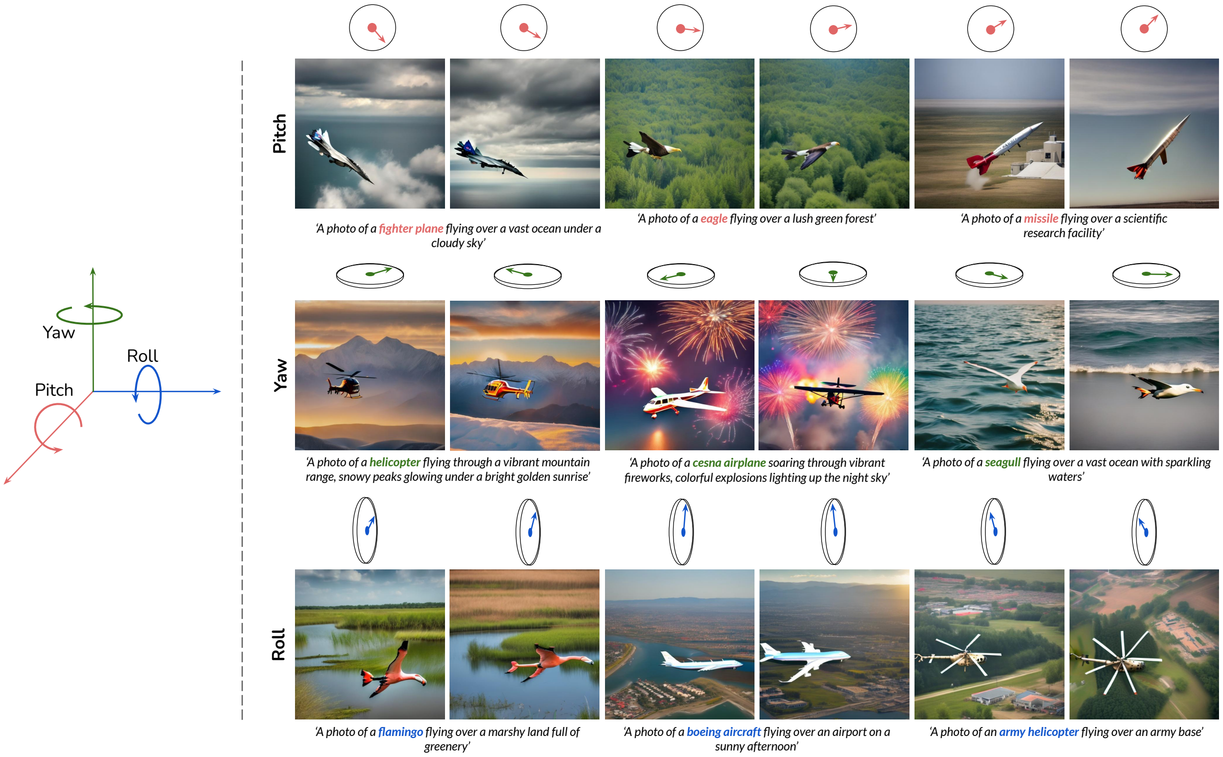

Yaw
Roll
Pitch
Pitch
'A photo of a fighter plane flying over a vast ocean under a cloudy sky'
'A photo of a eagle flying over a lush green forest'
'A photo of a missile flying over a scientific research facility'
Yaw
'A photo of a helicopter flying through a vibrant mountain range, snowy peaks glowing under a bright golden sunrise'
'A photo of a cesna airplane soaring through vibrant fireworks, colorful explosions lighting up the night sky'
'A photo of a seagull flying over a vast ocean with sparkling waters'
Roll
'A photo of a flamingo flying over a marshy land full of greenery'
'A photo of a boeing aircraft flying over an airport on a sunny afternoon'
'A photo of an army helicopter flying over an army base'

Figure 14. Conditioning on all three orientation angles

StableDiffusion-XL backbone model shown in Fig. 15. The results demonstrate improved image quality with accurate orientation control of the generated objects.

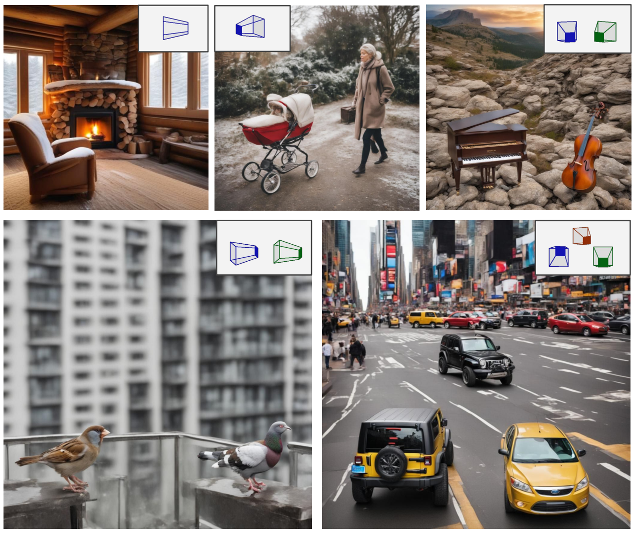


Figure 15. Compass Control on StableDiffusion-XL.

E. Diverse poses for non-rigid objects

We build our dataset with only a few synthetic objects in their fixed canonical pose to generate our training data. This makes the model prone to overfitting on these poses for the non-rigid objects in the dataset - dog, horse, and lion. For instance, during inference, the model can generate only standing dogs in the given orientation. We generate augmentations with realistic pose variations in the training data to mitigate this. Specifically, we randomly mask some regions from the Canny Edge map and pass it to the ControlNet (Fig. 16a). This allows ControlNet to freely generate any plausible pose within the masked region. When trained with resulting augmentations, our method can generate diverse pose variations of non-rigid training objects while following the precise orientation as shown in Fig. 16b).

F. Robustness to the 2D bounding boxes

Coarse bounding boxes. We analyze the robustness of required 2D object bounding boxes during the inference. First, we analyze the effect of the coarseness of the bounding box on the generated scenes in Fig. 17. Our model does not generate objects that tightly occupy the provided bounding box. This is convenient for the user, as they don't have to provide an exact 2D bounding box. We present results for different bounding box sizes while keeping the center fixed.

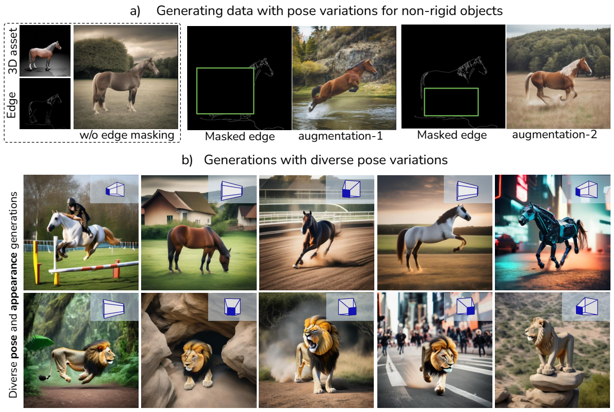

a) Generating data with pose variations for non-rigid objects
3D asset
Edge
w/o edge masking
Masked edge
augmentation-1
Masked edge
augmentation-2
b) Generations with diverse pose variations
Diverse pose and appearance generations

Figure 16. Pose variations for non-rigid objects

The model is robust to size changes and generates realistic scene compositions. The objects fall inside the box but they don't tightly fit the box. This provides more flexibility to the base generative model in generating more realistic scenes with relaxed constraints than conditioning on precise bounding boxes.

Spawning random boxes. In another experiment, we randomly spawn non-overlapping boxes, eliminating the user requirement to provide 2D boxes. The results are present in Fig. 17. Our method generates realistic compositions for these random layouts, with precise orientation control. The proposed design of using *loose* bounding boxes during training, enables this, as the objects can adjust their size within the box region to make coherent scenes.

Overlapping boxes. We present an ablation with the amount of overlap of 2D object boxes in Fig. 18 during inference. Our method can handle the overlap between 2D boxes upto a good extent. On increasing the overlap the models' performance gracefully degrades in the pose control as the overlapping region is controlled by both the pose tokens (jeep in 4th example). With a large overlap in the bounding boxes, the model fails to generate both objects, and this is one of the limitations of our proposed approach, which is based on attention regularization. However, this limitation is common across all the bounding box conditioned or guided generative models.

G. Discussion with SoTA object-centric works.

We compare the framework of our approach with recent works on object-centric 3D control in generation and editing with diffusion models. Particularly, we contrast our method with Neural Assets [63] and LooseControl [4], as these two are the closest method to ours. We present a comparison with both these methods at an approach level in Tab. 3.

H. Synthetic data generation

We render scenes with 3D assets in a Blender [15] environment for our dataset. Specifically, we place an opaque floor on the $x-y$ plane and place a camera tilted slightly towards the ground at a fixed position. The scene is lighted using 3 point lights of random intensity, placed at random locations. Once the environment is ready, we place the 3D assets at random locations and orientations and render the scene. For each rendered image we store the identity of the 3D assets in it, their respective orientations and 2D bounding boxes. We constrain the locations and orientations so that the object completely lies within the rendered image. Additionally, for two object scenes, we ensure that their 2D bounding boxes do not overlap. In all, we have 1000 one-object scenes and 7900 two-object scenes. Some samples from the rendered images can be found in Fig. 19.
However, training on this dataset alone leads to over-fitting to the plain backgrounds, as we have presented in the ablative experiments in the main text. Therefore, to generate the objects in diverse contexts, we augment the rendered scenes using Canny ControlNet [68]. Specifically, given a rendered scene, we extract it's Canny map using OpenCV [6], with the low and high thresholds set to 100 and 200 respectively. We use the following prompts for the augmentations:

1. a photo of $\langle subject \rangle$ in a snowy forest, with a gentle snowfall and snow-covered trees
2. a photo of $\langle subject \rangle$ in a vast desert with towering sand dunes and a clear blue sky
3. a photo of $\langle subject \rangle$ in a medieval castle courtyard with ancient stone walls and archways
4. a photo of $\langle subject \rangle$ in a sunflower field under a clear blue sky
5. a photo of $\langle subject \rangle$ in a dense rainforest, with sunlight streaming through the canopy
6. a photo of $\langle subject \rangle$ in a serene Japanese garden, surrounded by cherry blossoms
7. a photo of $\langle subject \rangle$ on a rocky cliff overlooking a vast ocean
8. a photo of $\langle subject \rangle$ by a riverside with wildflowers blooming nearby
9. a photo of $\langle subject \rangle$ at a river's edge with stones scattered around
10. a photo of $\langle subject \rangle$ in front of the Eiffel Tower at sunset
11. a photo of $\langle subject \rangle$ in a vibrant autumn forest, with orange and red leaves carpeting the ground
12. a photo of $\langle subject \rangle$ in a vast open plain, with golden grasses swaying in the wind and distant mountains on the horizon under a wide, clear sky
13. a photo of $\langle subject \rangle$ on a cobblestone street in a quaint European village, with flower-filled balconies and historic buildings
14. a photo of $\langle subject \rangle$ in a canyon with towering red rock

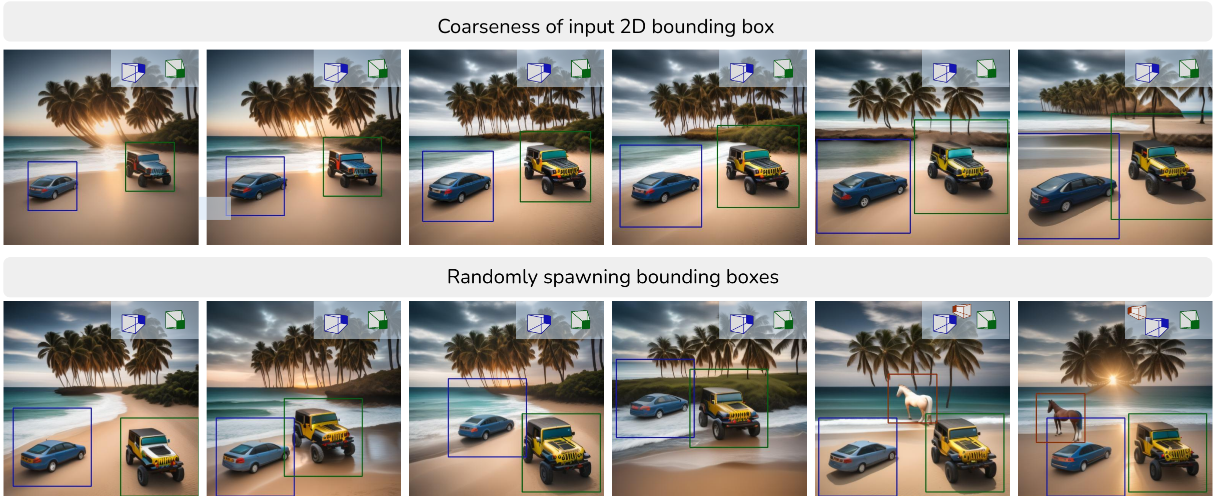

Coarseness of input 2D bounding box
Randomly spawning bounding boxes

Figure 17. **Robustness of 2D bounding boxes.** Our method generates realistic scene compositions with different 2D bounding box sizes. Allowing for a loose bounding box during training provides this flexibility to the model to generate realistic scenes while coarsely following the input 2D box. Further, random non-overlapping boxes can also be spawned during inference without any degradation in quality. This robustness to the actual bounding box shape, reduces the burden on the user and is enabled by the *loose* bounding box used during training.

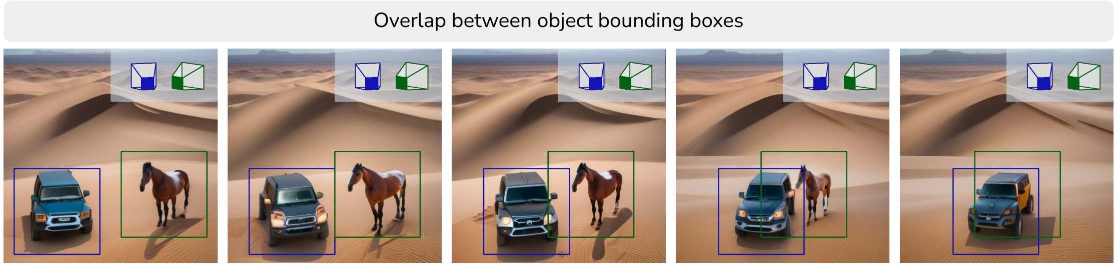


Figure 18. **Overlapping bounding boxes.** Our method can handle overlap between the two input bounding boxes up to a good extent. However, with a large overlap, the model struggles to generate accurate orientations (jeep in the fourth example), due to the mixing of pose tokens.

formations, and scattered desert plants growing in the rocky terrain

We run this augmentation pipeline on all the rendered images, and do a manual filtering to remove the inconsistent generations. In all, we have 771 single-object augmentations and 5239 two-object augmentations.

I. Orientation Regressor

We train a neural network model to predict the orientation angle of an object in the generated image. We use a pre-trained ResNet-18 [23] as the feature extractor and a mlp head consisting of two hidden layers of 128 neurons, each with ReLU activations. Finally, we predict a single orientation angle θ along the up-axis (details in the main text - sec.3.1). We call this model *orientation regressor* and train with a dataset of $35K$ images generated by rendering 30 synthetic 3D assets of the test object categories followed by their canny ControlNet augmentations. This data is highly diverse, containing various backgrounds and object appearances, enabling the learning of an accurate orientation regressor. We train with a batch size of 128, a learning rate of $5e-5$ for 95 epochs with Adam optimizer. On an unseen test set of $8K$ images from the same distribution, the trained model achieves a mean angular error of 0.125. Further, we present the results for evaluation on a completely unseen dataset, generated by Stable Diffusion [51], containing the test objects in Fig. 20. We can observe that the trained orientation regressor predicts accurate orientations, and hence, it is a good estimator for evaluating pose consistency. In the case of multi-object scenes, we crop out the objects using Grounding DINO [36] and pass them to the *orientation regressor*.

	Model type	Training data	Input during inference	Novel categories	Input Representation	Personalization
LooseControl [4]	Generation	Real images (w 3D boxes)	3D object boxes	Yes	Explicit 3D (Depth)	No
Neural Assets [63]	Editing	Real videos (w 3D boxes)	3D object boxes	No	Implicit (List of bbox)	Yes
Ours	Generation	Synthetic images (w Orientation + 2D boxes)	Orientation + 2D object boxes	Yes	Implicit List of orientations	Yes

Table 3. Comparison with state-of-the-art approaches for object-centric control in the generation process.

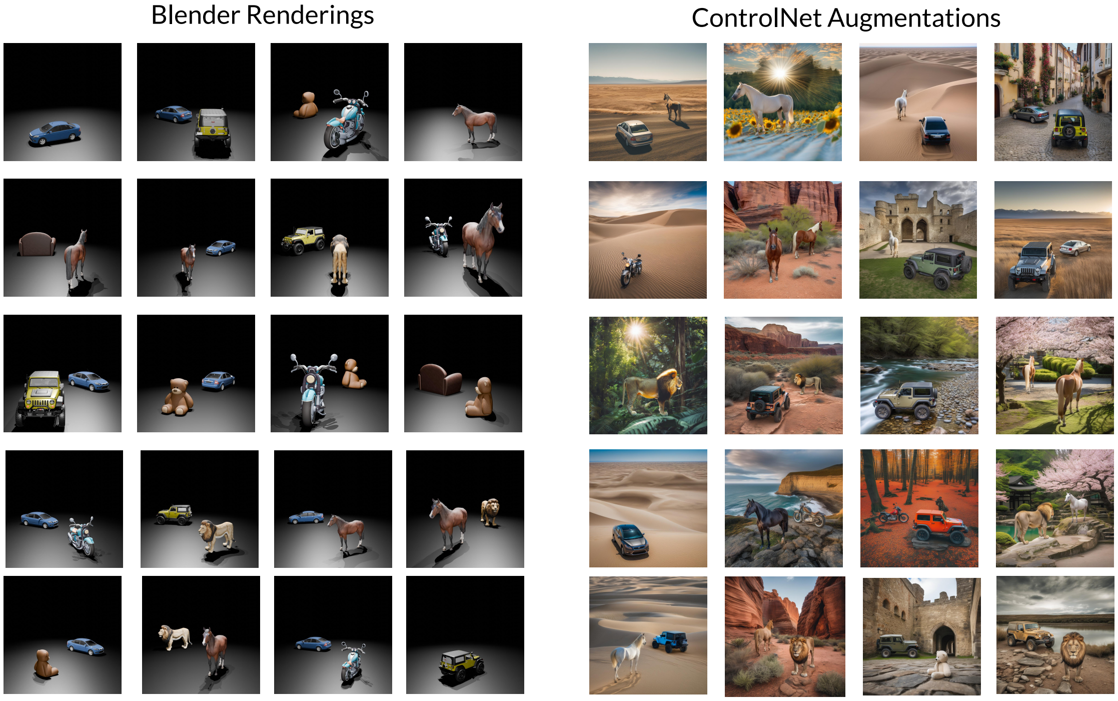


Figure 19. Samples from data generation process

J. Baseline details

We provide implementation details for the baselines discussed in the paper.

J.1. ViewNeTI [7]

ViewNeTI trains a small MLP to project the 3D camera pose to 3D view token. This token, along with the text prompt, is used to condition the text-to-image model. In the basic form, it is trained on a single scene with multi-view images and 3D camera poses. Once trained, the model can generate novel views for the trained scene. However, in an extended version, it is trained with multiple scenes to learn a generalizable view token. This token is then used for view control in text-to-image generation. For comparison, we use this version and train on our synthetic dataset of rendered multi-view scenes. Specifically, instead of conditioning on 3D camera pose, we condition on orientation angle θ and predict the view token. We train the model for $60K$ iterations on 1000 multi-view images of 10 assets. Note that because this model only supports a global view control, we train and evaluate it on only single object scenes for orientation control.

J.2. Continous 3D Words [14]

In this approach, a text-to-image diffusion model is conditioned on continuous 3D tokens to control 3D attributes such as lighting and object pose. They learn a generalizable *3D word* in the text embedding space of the T2I model for each attribute, which is used along with the text prompt for conditioning. To learn the 3D word token, they use renderings of a single object and generate its augmentations with depth-conditioned ControlNet. However, it is essential that the 3D word token is disentangled from the object used for training. For this, they follow a staged training procedure: first learn the object's appearance (stage 1), and then learn the 3D attribute (stage 2). Following this, we train this

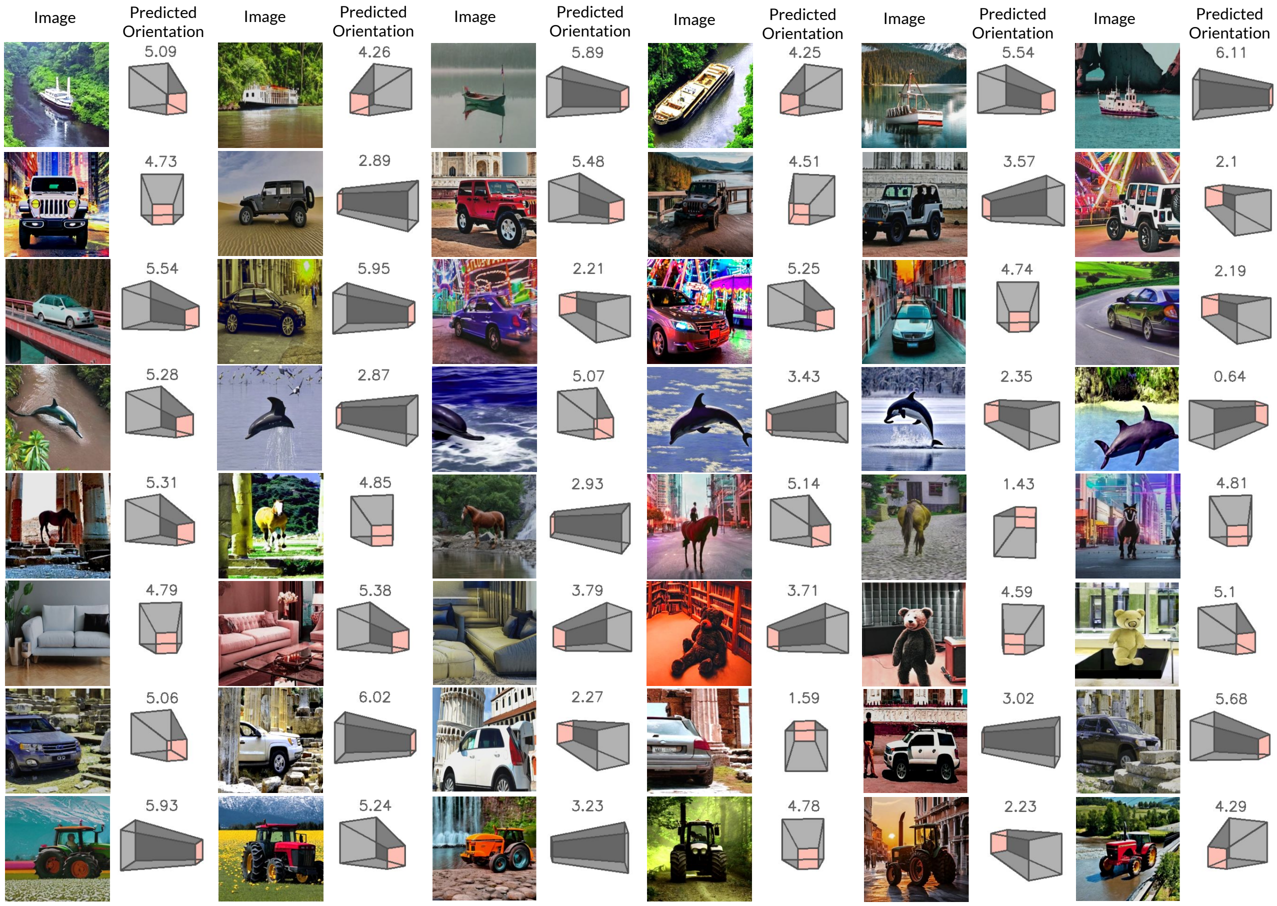


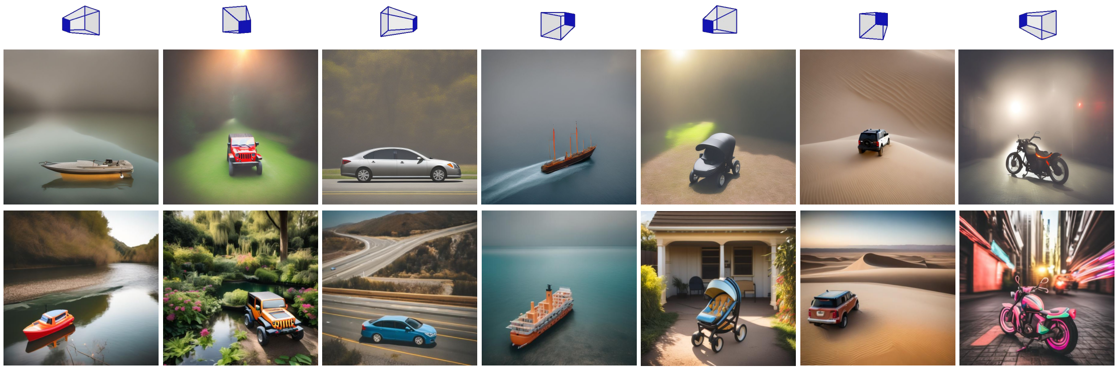
Figure 20. Predictions of the trained orientation regressor on unseen samples generated from Stable Diffusion [51]. The model can predict the orientations accurately for the diverse unseen data and acts as a good critic to evaluate orientation consistency in generated images.

model a single 3D asset, *sedan*. We train for 5000 iterations in stage 1 and 15000 iterations in stage 2 (same as the original model). However, the trained model poorly generalizes to new objects as it is trained on a single object mesh (Fig.7 in the main text).

Here we present a variant of this model, which is trained on multiple 3D assets instead of just one (as proposed in their original paper). We use the same rendered images dataset as ours, and augment it using their proposed augmentation strategy. Notably, this dataset has diverse layouts and objects placed at random locations in the scene, making the learning process challenging. Since this model only allows for global control, we train and evaluate it on single object scenes only. We perform 30000 training iterations in the first stage to learn the object appearance, followed by 70000 iterations to learn the 3D word token. The comparison is presented in Fig. 21. Our method achieves superior performance as compared to this baseline. The baseline struggles in pose control due to high diversity in the scene layouts, highlighting the importance of our attention localization mechanism CALL. Further, our backgrounds are much richer, as we use canny-conditioned ControlNet augmentations, which leads to richer augmentations.

J.3. LooseControl [4]

LooseControl is a conditioning framework on text-to-image diffusion models that allows for 3D scene layout control. The framework is built on a depth-conditioned ControlNet model. However, instead of relying on accurate depth maps, which are often difficult to construct, LooseControl conditions the generation on coarse depth maps. Specifically, in this loose depth map, the scene boundaries are represented as planes, and the objects are represented by their loose 3D bounding boxes. LooseControl is implemented as a LoRA [26] fine-tuning over depth-conditioned ControlNet model. This fine-tuning enables it to condition the generation using loose object depth maps also, against the accurate depth maps required by original ControlNet. In



'A photo of a **boat** floating in a river'

'A photo of a **jeep** in a garden'

'A photo of a **sedan** on a highway'

'A photo of a **ship** floating in the sea'

'A photo of a **pram** in the backyard of a house on a sunny afternoon'

'A photo of a **SUV** in a desert with towering sand dunes'

'A photo of a **motorbike** in an urban city street with neon lights'

Figure 21. Comparsison with modified Continuous 3D Words [14] trained on multiple assets. Compass control generates more realistic outputs and follows the text prompt better than the Cont-3D-Words trained on multiple object datasets.

our experiments, we generate the loose depth maps by placing 3D bounding boxes in a Blender [15] environment, and rendering the depth from camera viewpoint. Specifically, we randomly sample objects' locations and pose within the scene boundary and place a 3D bounding box for each object. Notably, one can control the object orientation by rotating the corresponding 3D bounding box in the input. We define a fixed template of 3D bounding box dimensions for each test object in the dataset. The obtained depth maps are used to condition the model. We used the publicly available checkpoint for LooseControl in our evaluation. As this method allows for multi-object control, we compare both single and multi-object scenes. However, in experiments, we observe that LooseControl struggles to generate multi-object scenes with precise pose control and often resorts to generating bounding box artifacts. This is primarily due to the strong depth conditioning prior in the base depth ControlNet model, which is trained to follow exact depth maps.

K. Additional Results

K.1. Comparisons

We present additional baseline comparison results in Fig. 23. Our method follows the text prompts and generates objects following the input prompts

L. Implementation Details

L.1. Method details

We use Stable Diffusion v2.1 [51] as our base T2I model and use LoRA rank 4 for fine-tuning its UNet. Our encoder model $\mathcal{P}$ is a lightweight MLP: three linear layers with ReLU. We train our model for $100K$ iterations with a batch size of 4 with AdamW optimizer and a fixed learning rate of 10^{-4}. We train first stage for $30K$ iterations with only single object scenes and the next stage for $70K$ iterations with mix of single and two subject scenes. We use SD-Xl for generating augmentations due to its higher realism.

We keep the bounding box padding $\lambda = 1.2$ for CALL. The training takes 24 hours on a single A6000 GPU, thus highly efficient.

L.2. Evaluation dataset

We randomly sample 10 pose orientation in the range of (0,360 deg) for each prompt and object combination. We used the following set of prompts for evaluation, containing single and two subject. In each prompt *¡subject¿* is replaced with a single subject (e.g., jeep) or two subjects (e.g., jeep and sedan). Notably these prompts are different that the one used to generate ControlNet augmentations, to accurately evaluate model generalization.

For road objects

1. A photo of $\langle subject \rangle$ in front of the Taj Mahal
2. A photo of $\langle subject \rangle$ on the streets of Venice, with the sun setting in the background
3. A photo of $\langle subject \rangle$ in front of the leaning tower of Pisa in Italy
4. A photo of $\langle subject \rangle$ in a modern city street surrounded by towering skyscrapers and neon lights
5. A photo of $\langle subject \rangle$ in an ancient Greek temple ruin, with broken columns and weathered stone steps
6. A photo of $\langle subject \rangle$ in a field of dandelions, with snowy mountain peaks in the distance
7. A photo of $\langle subject \rangle$ in a rustic village with cobblestone streets and small houses
8. A photo of $\langle subject \rangle$ on a winding country road with green fields, trees, and distant mountains under a sunny sky
9. A photo of $\langle subject \rangle$ in front of a serene waterfall with

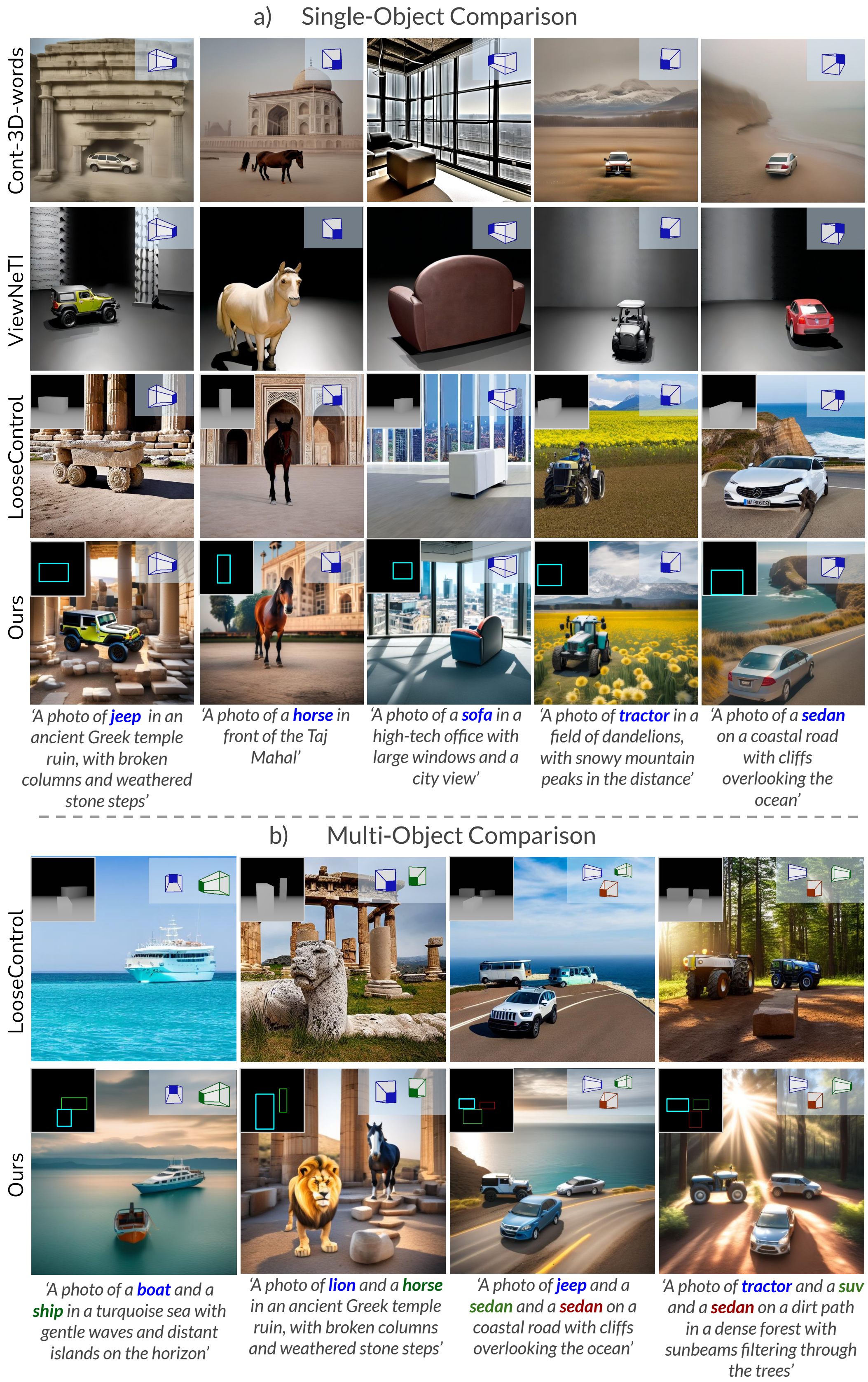
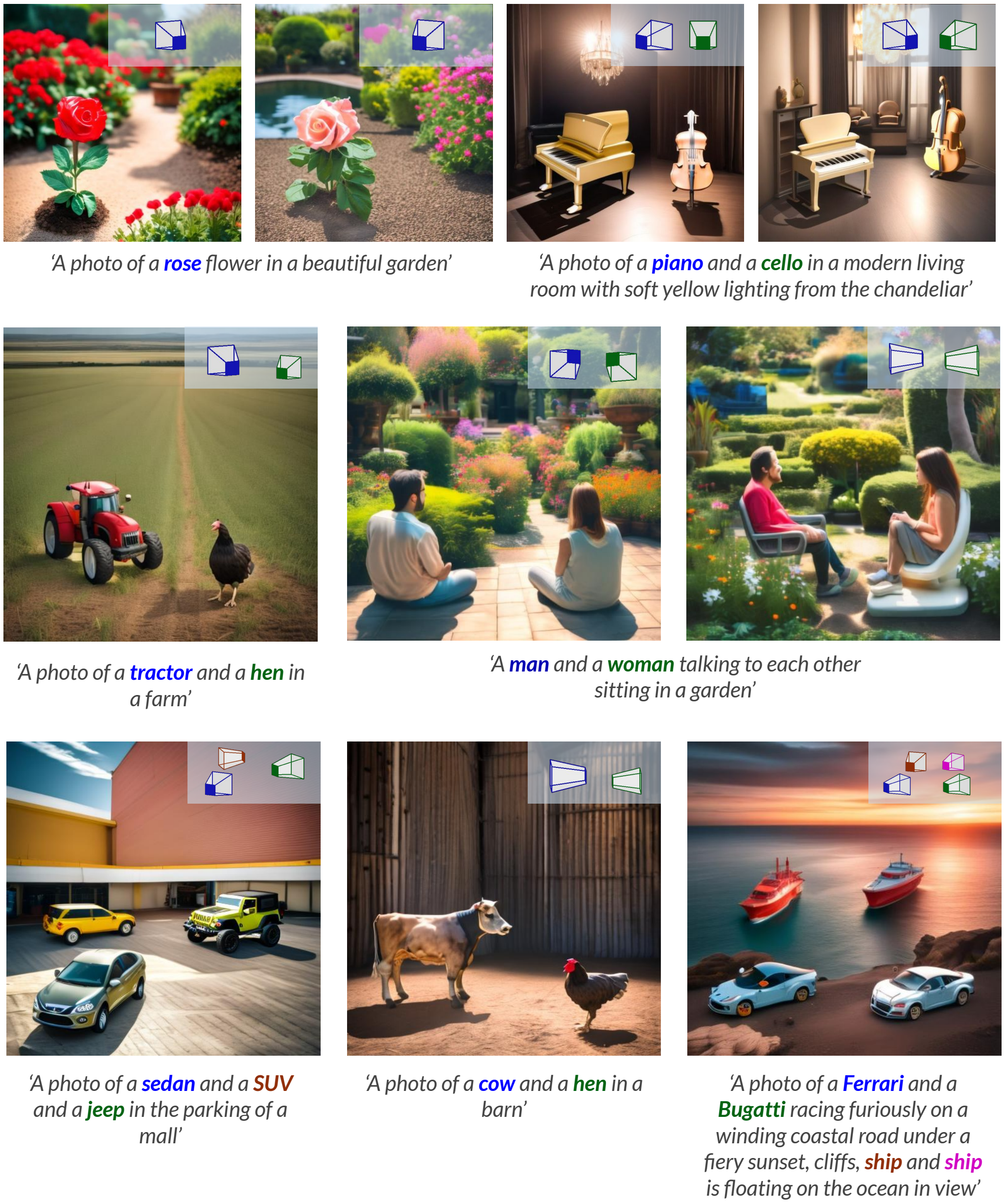

a) Single-Object Comparison
Cont-3D-words
ViewNeTI
LooseControl
Ours
'A photo of jeep in an ancient Greek temple ruin, with broken columns and weathered stone steps'
'A photo of a horse in front of the Taj Mahal'
'A photo of a sofa in a high-tech office with large windows and a city view'
'A photo of tractor in a field of dandelions, with snowy mountain peaks in the distance'
'A photo of a sedan on a coastal road with cliffs overlooking the ocean'
b) Multi-Object Comparison
LooseControl
Ours
'A photo of a boat and a ship in a turquoise sea with gentle waves and distant islands on the horizon'
'A photo of lion and a horse in an ancient Greek temple ruin, with broken columns and weathered stone steps'
'A photo of jeep and a sedan and a sedan on a coastal road with cliffs overlooking the ocean'
'A photo of tractor and a suv and a sedan on a dirt path in a dense forest with sunbeams filtering through the trees'

Figure 22. Additional comparison results with the baselines for single object and multi-object scenes.


'A photo of a **rose** flower in a beautiful garden'

'A photo of a **piano** and a **cello** in a modern living room with soft yellow lighting from the chandeliar'

'A photo of a **tractor** and a **hen** in a farm'

'A **man** and a **woman** talking to each other sitting in a garden'

'A photo of a **sedan** and a **SUV** and a **jeep** in the parking of a mall'

'A photo of a **cow** and a **hen** in a barn'

'A photo of a **Ferrari** and a **Bugatti** racing furiously on a winding coastal road under a fiery sunset, cliffs, **ship** and **ship** is floating on the ocean in view'

Figure 23. More qualitative results from our method, *Compass Control*.

trees scattered around the region, and stones scattered in the water

10. A photo of $\langle subject \rangle$ on a sandy desert road with dunes and a vast, open sky above
11. A photo of $\langle subject \rangle$ on a bridge overlooking a river with mountains in the background
12. A photo of $\langle subject \rangle$ on a dirt path in a dense forest with sunbeams filtering through the trees
13. A photo of $\langle subject \rangle$ on a coastal road with cliffs overlooking the ocean

14. A photo of $\langle subject\rangle$ in front of a historical castle with high stone walls and flags flying in the breeze
15. A photo of $\langle subject\rangle$ in front of an amusement park with bright lights and ferris wheels in the background

For water objects

1. A photo of $\langle subject\rangle$ on still waters under a cloudy sky, mountains visible in the distant horizon
2. A photo of $\langle subject\rangle$ floating on a misty lake, surrounded by calm waters and serene, foggy atmosphere
3. A photo of $\langle subject\rangle$ in the vast sea, with a clear blue sky and a few fluffy clouds
4. A photo of $\langle subject\rangle$ in the middle of a stormy ocean, with dark clouds and crashing waves
5. A photo of $\langle subject\rangle$ in a calm lake with lily pads and reeds growing near the shoreline
6. A photo of $\langle subject\rangle$ on a river running through a dense jungle with vibrant green foliage
7. A photo of $\langle subject\rangle$ in a mountain lake surrounded by pine trees and snow-capped peaks
8. A photo of $\langle subject\rangle$ floating in a lagoon with tropical fish and coral visible beneath the water
9. A photo of $\langle subject\rangle$ on a frozen lake with a snowy landscape surrounding it
10. A photo of $\langle subject\rangle$ on a serene river at dusk, with reflections of the sunset on the water
11. A photo of $\langle subject\rangle$ in the middle of a vast marshland with tall grasses and migratory birds flying overhead
12. A photo of $\langle subject\rangle$ near a small waterfall cascading into a clear pool in a rocky area
13. A photo of $\langle subject\rangle$ on a bay with large rock formations jutting out of the water
14. A photo of $\langle subject\rangle$ in a turquoise sea with gentle waves and distant islands on the horizon
15. A photo of $\langle subject\rangle$ in a narrow canal in an old European city, with historic buildings lining the waterway

For indoor objects

1. A photo of $\langle subject\rangle$ in a modern living room setting with painted walls and glass windows
2. A photo of $\langle subject\rangle$ in a minimalist living room
3. A photo of $\langle subject\rangle$ in a cozy library with shelves filled with books and warm lighting
4. A photo of $\langle subject\rangle$ in a high-tech office with large windows and a city view
5. A photo of $\langle subject\rangle$ in an art studio with canvas paintings and art supplies scattered around
6. A photo of $\langle subject\rangle$ in a rustic kitchen with wooden cabinets and a stone countertop
7. A photo of $\langle subject\rangle$ in a lavish living room with elegant decor and soft lighting
8. A photo of $\langle subject\rangle$ in a large dining hall with chandeliers and long tables
9. A photo of $\langle subject\rangle$ in a traditional Japanese tatami room with sliding paper doors
10. A photo of $\langle subject\rangle$ in a well-equipped gym with weights and fitness machines
11. A photo of $\langle subject\rangle$ in a music studio with soundproof walls and musical instruments
12. A photo of $\langle subject\rangle$ in a sunlit greenhouse filled with tropical plants
13. A photo of $\langle subject\rangle$ in a children's playroom with colorful toys and posters on the walls
14. A photo of $\langle subject\rangle$ in an underground wine cellar with wooden barrels and dim lighting
15. A photo of $\langle subject\rangle$ in a cozy reading nook with a soft armchair and a small lamp