Multitwine: Multi-Object Compositing with Text and Layout Control

Supplementary Material

1. Training and Testing Data

In this section, we provide additional information about the training data generation pipeline proposed in Main Paper Section 3.4 and MultiComp, the multi-object compositing test set introduced in Main Paper Section 4.

1.1. Training Data

As detailed in Main Paper Section 3.4, our paired training data — comprising ground truth images with multiple objects, descriptive captions with grounding information, segmented images of two objects, and corresponding object-specific bounding boxes — is collected from complementary sources: Video Data, In-the-Wild Images, and Manually Collected Data. Below, we expand on how paired data is extracted from each source.

Video Data Fig 1 illustrates the process for extracting paired training data from videos in [36, 37]. A ground-truth frame containing an annotated relationship between two objects is randomly selected. Two additional frames, each showing one of the interacting objects, are extracted from the same video, ensuring a similarity between the views of each object (DINO score MSE $\geq$ 0.8) [27]. A caption describing the relationship is automatically generated by feeding the ground-truth image and annotated relation into LLaVA v1.6 (34B) [24] with the prompt: *"Can you provide a grammatically correct one-line caption for the relation <object A> <relation> <object B> in the image?"*. The segmented objects are then extracted using an off-the-shelf semantic segmentation model [31].

Image Data We propose two automated approaches for obtaining paired data from in-the-wild images.

Top-down approach As illustrated in Fig 2, this approach generates paired training data from a single image using a systematic process. First, a commercial subject selection tool identifies the main objects in the scene. A semantic segmentation model [31] then segments the selected region to determine the number of objects present. If multiple objects are detected, two are randomly selected as composited objects. Their outlines are highlighted on the image using distinct colors (e.g., orange and blue) and passed to ViP-LLaVA (13B) [3] for caption generation through two sequential prompts. In the first step, entities within each highlighted outline are identified with a question like: *"Please follow the sentence pattern of the example to list the entities within each rectangle. Example: 'orange: banana; blue: apple'"*. This produces responses such as *"orange: teddy bear; blue: girl"*. Using these entity labels, the second step generates a descriptive caption with a prompt like: *"Can you provide a one-line caption including the interaction between 'teddy bear within the orange rectangle' and 'girl within the blue rectangle' in the image by using these exact entity names?"*. The resulting caption, *e.g.*, *"A young girl within the blue rectangle is holding a large teddy bear within the orange rectangle"*, is refined by removing the grounding phrases *"within the orange/blue rectangle"* after using them for correlating object images with text tokens. This method ensures accurate grounding information, even when both entities are labeled the same, resulting in the final caption: *"A young girl is holding a large teddy bear"*.

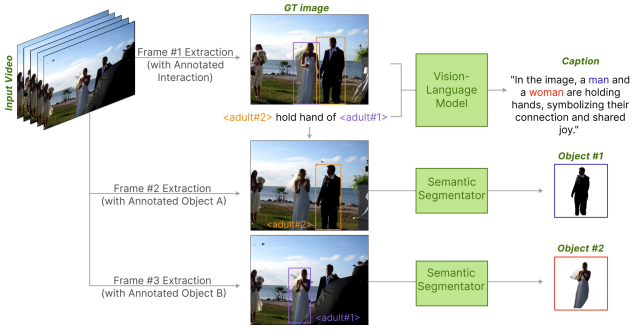

Figure 1. Training Data Generation from Video Data. Paired training data is obtained from video object relation datasets [36, 37] by extracting three frames with corresponding annotations and leveraging Vision-Language Models [24].

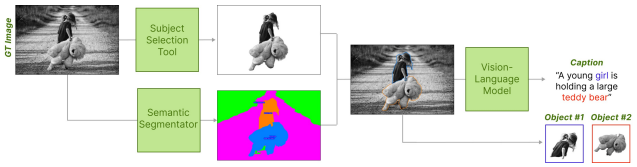

Figure 2. Training Data Generation from Image Data via Top-Down Approach. Paired training data is derived from in-the-wild images by leveraging a Vision-Language Model [3] and a Semantic Segmentator [31].

Bottom-up approach This approach (Fig 3) leverages a grounding model like GroundingDINO [25] to process paired ground truth images and captions. The model extracts bounding boxes that link specific words in the caption to objects in the image. Duplicates, background elements, and undesired objects (*e.g.*, overly large or small objects, or those with low confidence scores) are removed, leaving a set of object candidates with corresponding grounding information. Two of those objects are then randomly selected, and an off-the-shelf semantic segmentation model [31] is

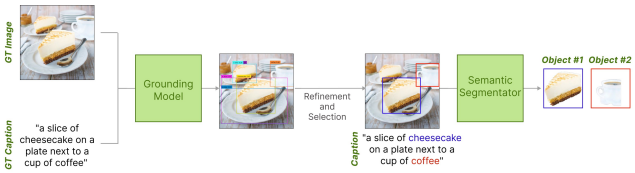

Figure 3. Training Data Generation from Image Data via Bottom-Up Approach. Paired training data is extracted from in-the-wild images with a paired caption by leveraging a Grounding Model [25] and a Semantic Segmentator [31].

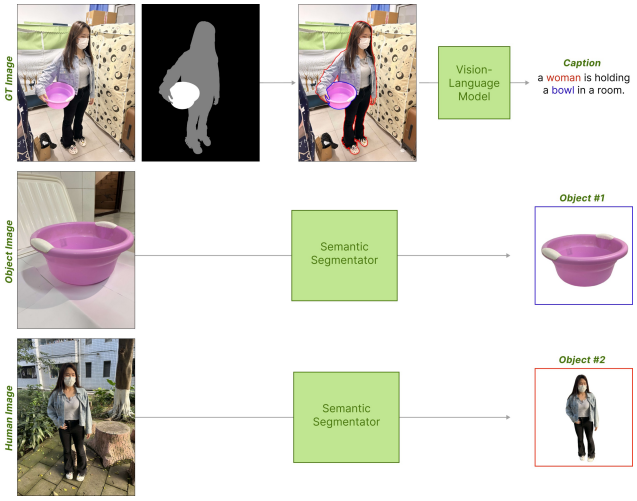

Figure 4. Training Data Generation from Manually Collected Data. Paired training data is obtained from a collected dataset containing object images, human images, and images of humans interacting with objects. We leverage a Vision-Language Model [3] and a Semantic Segmentator [31] to extract segmented objects and corresponding caption with grounding information.

used to extract them. This results in two segmented objects along with their associated grounding details from the original caption.

Manually Collected Data We manually collect and annotate images featuring objects, humans, and human-object interactions. For captioning with grounding information, we use ViP-LLaVA (13B) [3], following the exact same procedure as the top-down approach described above. Additionally, a semantic segmentation model [31] is employed to segment entities in images containing either a single object or a human. Fig 4 illustrates this approach.

2. Inference Data

Our collected MultiComp set consists of 119 paired data entries, each containing: (i) a background image, (ii) two object images, (iii) object-specific bounding boxes, (iv) an inpainting bounding box encompassing the previous ones, and (v) a descriptive caption with grounding information. Background images are sourced from Pixabay [29], while objects are from Pixabay [29], MultiBench [23], and DreamBooth [34]. Bounding boxes and captions are manually crafted. For evaluation, we perform 5 iterations of each model on the entire set, resulting in 595 generated images.

Out of these, 395 images contain overlapping bounding boxes for the two objects (MultiComp-*overlap*), while 200 show non-overlapping bounding boxes (MultiComp-*nonoverlap*). We evaluate these subgroups separately due to their differing levels of difficulty. While simultaneous compositing offers benefits like cohesive harmonization in both cases, it is especially effective in the overlapping cases, where it allows for simultaneous object reposing and the generation of additional elements needed for the scene.

When textual input is provided, we further categorize the set into two subgroups based on caption types: MultiComp-*action* and MultiComp-*positional*. The MultiComp-*action* subset contains 375 images generated from action-based captions (*e.g.*, *running after, playing with, holding*), while the MultiComp-*positional* subset includes 220 images generated from captions describing positional relations (*e.g.*, *next to, behind, in front*). This distinction allows us to separately evaluate cases where reposing objects is often necessary (MultiComp-*action*) versus those primarily describing object layout (MultiComp-*positional*).

3. Comparison to Existing Methods

We provide additional visualizations comparing our model to existing generative object compositing and multi-entity subject-driven generation models in Sections 3.1 and 3.2. Further details on user studies can be found in Section 3.3.

3.1. Comparison to Generative Object Compositing Methods

We visually compare our simultaneous multi-object compositing method to sequentially adding two objects using State-of-the-Art Generative Compositing Methods [7, 40, 43, 48, 50] in Fig 5.

3.2. Comparison to Subject-Driven Generation Methods

We visually compare two-entity subject-driven generation using our method and existing methods with available code (BLIP-Diffusion [22], KOSMOS-G [28] and Emu2Gen [42]) in Fig 6.

3.3. User Studies

We conduct six user studies to evaluate our multi-object compositing model against other generative object compositing models [7, 40, 43, 48, 50] and Emu2Gen [42]. In each study, non-expert users are shown two side-by-side images, one generated by our model and the other by a baseline, presented in random order. Users are asked to choose

the preferred image based on a specific criterion. The entire MultiComp set is used for each experiment, except for the 'most realistic interaction' evaluation, where only images from MultiComp-*overlap* are considered, as this subset best evaluates the task. At least five users rate each image pair, and the results are aggregated via majority consensus. Visual examples of each experiment and the specific questions posed to the users can be found in Figs 7, 8, 9, 10, 11, and 12.

4. Ablation Study

Fig 13 shows visual examples of images generated by each ablation of our model (as detailed in Main Paper Table 3) for the same set of inputs. Without multi-view data (*i.e.*, video data, manually collected data), the model struggles to properly repose and combine objects to align with the textual description. In the absence of joint training for compositing and customization, the model fails to balance textual and visual inputs, resulting in object identity loss when reposing. Without cross-attention and/or self-attention losses, disentangling object identities becomes difficult, leading to texture and color leakage between objects (*e.g.* bow color, cat ear on ball). Lastly, omitting the masking step during inference leads to a degradation in alignment with the input layout and fails to fully mitigate identity leakage.

5. Applications

5.1. Model Versatility

We demonstrated in Main Paper Fig 9 how, by leveraging the advantages of our joint compositing and customization training, our model can be used for subject-driven inpainting. Additionally, Fig 14 illustrates how the same model can be applied to a broad range of tasks:

Layout-Driven Inpainting This task takes as input a descriptive caption, a background image, and a layout specifying an inpainting region along with object-specific bounding boxes for objects referenced in the caption. The model inpaints the selected region of the background image, ensuring alignment with the textual description while positioning objects according to the provided layout.

Multi-Object Compositing In addition to the inputs required for layout-driven inpainting, this task includes an image for each object corresponding to the provided bounding boxes. The model maintains the identity of these objects while enabling reposing and view synthesis, producing a cohesive composited image.

Layout-Driven Generation In this case, no background image is provided. This task uses only a descriptive caption and bounding boxes specifying object positions as inputs. The model generates a full image that aligns with the caption while placing objects in the specified locations.

Multi-Entity Subject-Driven Generation Similar to layout-driven generation, this task uses a text caption and bounding boxes as inputs but also includes an image for each object. The model generates a complete scene that aligns with the text, places objects in their specified locations, and preserves their unique identities.

5.2. Multi-Object Compositing and Multi-Entity Subject-Driven Generation

Fig 15 show how the same model can be used for both multi-object compositing and multi-entity subject-driven generation, guided by a variable number of provided objects.

5.3. Robust Object Compositing

During training, an off-the-shelf segmentation model (EntitySeg [31]) is used to extract single objects from images, a process that introduces segmentation errors and occlusions. These imperfections during training strengthen our model's ability to handle real-world scenarios at inference. As a result, Multitwine demonstrates robust compositing capabilities, seamlessly adapting object attributes to produce natural-looking results even in challenging scenarios. Fig 16 illustrates several examples of these capabilities, including: managing the transparency of a glass of milk, handling incomplete objects and inaccurate segmentations, and reharmonizing subjects from different domains (*i.e.* blending two cats extracted from a color image and a black-and-white image).

5.4. Robust Layout Alignment

Multitwine incorporates a layout cue as input, allowing users to specify the precise location and shape of each composited object. The model is trained to adapt object dimensions to fit the provided input masks, offering users greater control over the final output, as demonstrated in Fig 17. However, this strict bounding box guidance can sometimes lead to slight deformations in the objects, as commonly seen with generative compositing models. If deformations are undesirable, training with perturbed masks could mitigate them, though at the cost of some layout control.

5.5. Attribute Editing through Text

As seen in Fig 18, our model is designed to accommodate complex, multi-word grounded descriptions for each composited object, effectively capturing and reflecting their nuances in the final image. This capability ensures greater fidelity and detail in the generated results, aligning closely with the provided descriptions. In this case, multimodal embeddings are created by concatenating visual information after the last corresponding text token, while cross-attention masking and inference operate on all grounded text tokens.

6. Limitations and Failed Cases

Although simultaneously compositing several objects is possible with Multitwine, our model is not specifically designed for handling an unlimited number of objects. Fig 19 depicts a case where our model fails to composite six objects in a complex scene. When attempting to composite many objects, the model struggles to naturally integrate them all, resulting in missing objects, weaker harmonization, and reduced text-image alignment.

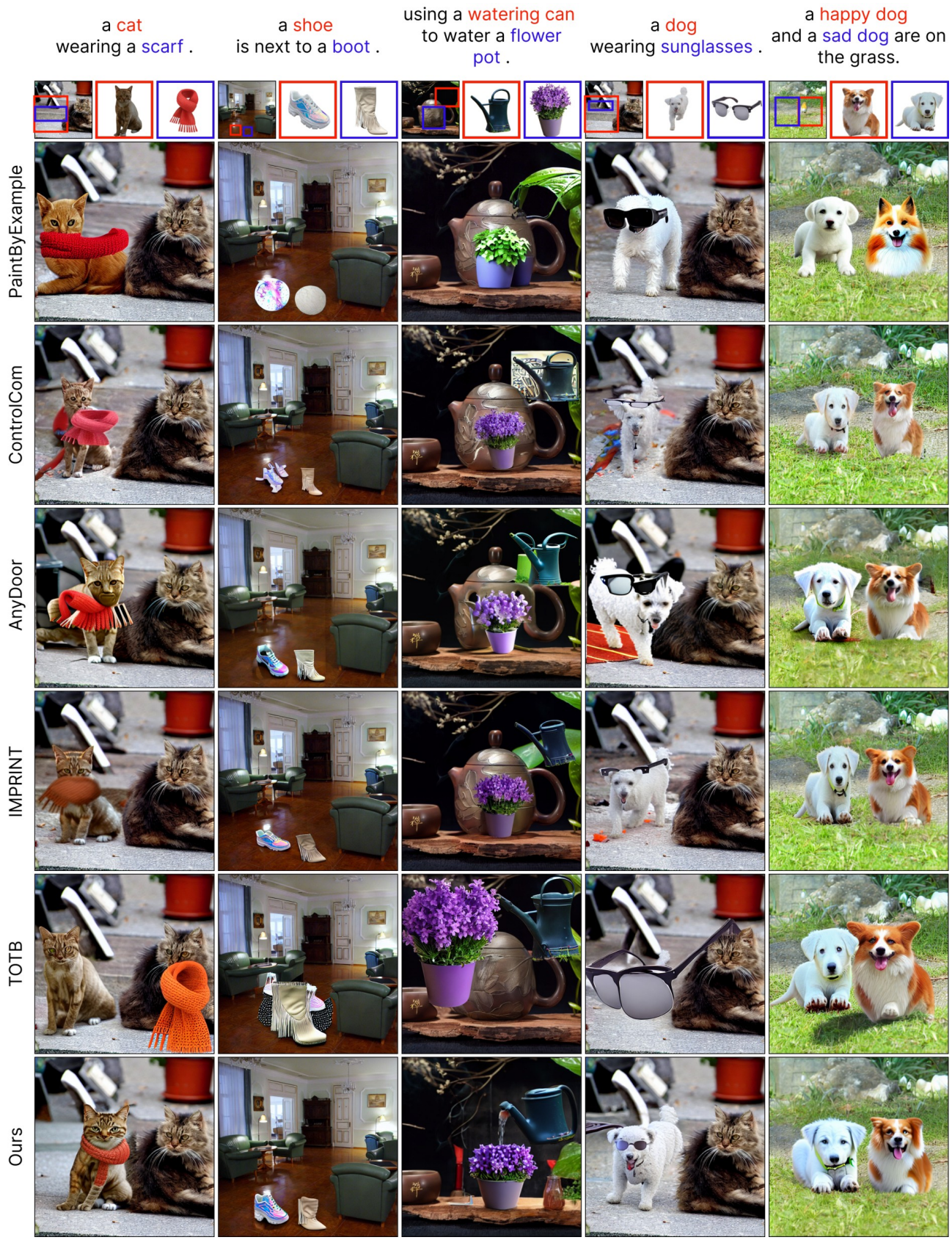

Figure 5. Visual comparison of our Multi-Object Compositing Method and State-of-the-Art Generative Object Compositing Methods [7, 40, 43, 48, 50].

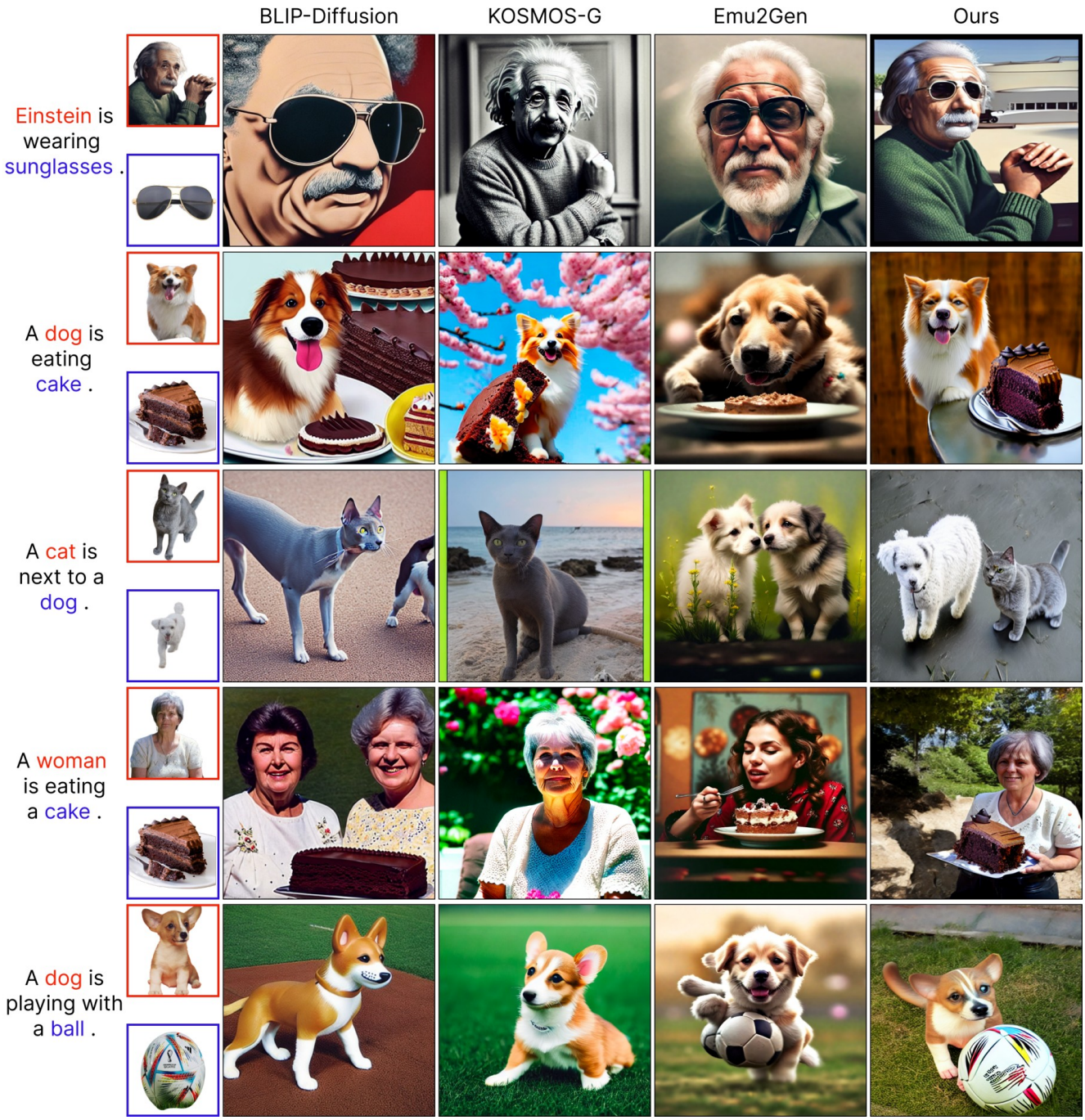

Figure 6. Visual comparison of our Customization Method and State-of-the-Art Subject-Driven Generation Methods [22, 28, 42].

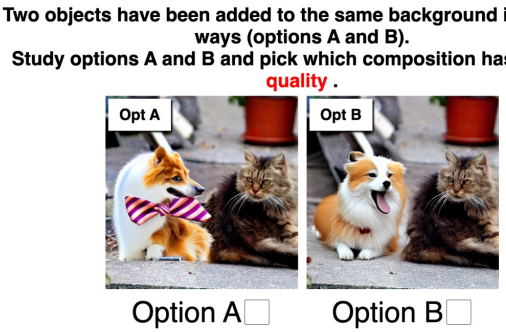

Figure 7. User Study on 'Compositing Quality'. Screenshot of user study presented to participants for evaluating the image quality of our multi-object compositing method against generative object compositing baselines [7, 40, 43, 48, 50].

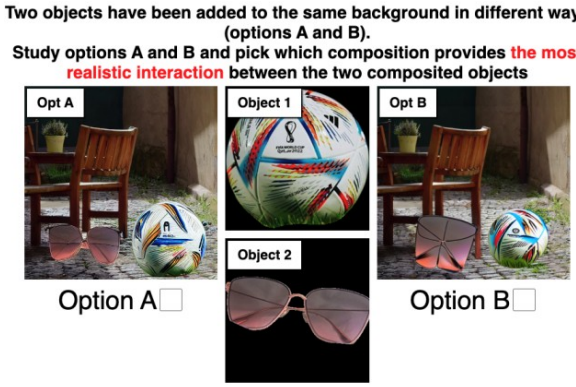

Figure 8. User Study on 'Realistic Interaction'. Screenshot of user study presented to participants for evaluating the realism of interactions generated by our multi-object compositing method against generative object compositing baselines [7, 40, 43, 48, 50].

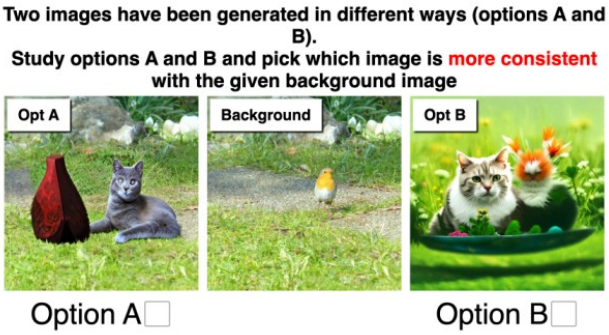

Figure 9. User Study on 'Background Alignment'. Screenshot of user study presented to participants for evaluating the alignment with background image of our multi-object compositing method against Emu2Gen [42].

Figure 10. User Study on 'Text Alignment'. Screenshot of user study presented to participants for evaluating the text alignment of our multi-object compositing method against Emu2Gen [42].

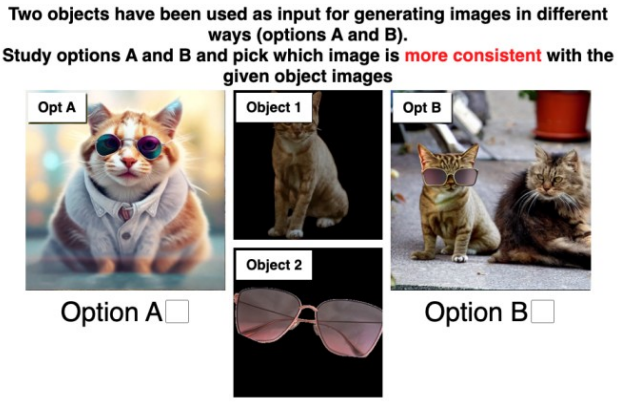

Figure 11. User Study on 'Objects Alignment'. Screenshot of user study presented to participants for evaluating the alignment with input object images of our multi-object compositing method against Emu2Gen [42].

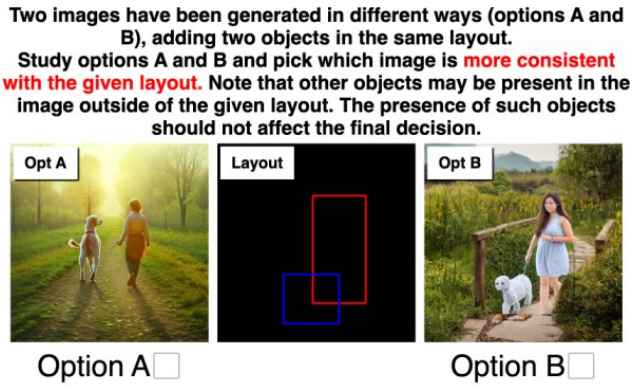

Figure 12. User Study on 'Layout Alignment'. Screenshot of user study presented to participants for evaluating the layout alignment of our multi-object compositing method against Emu2Gen [42].

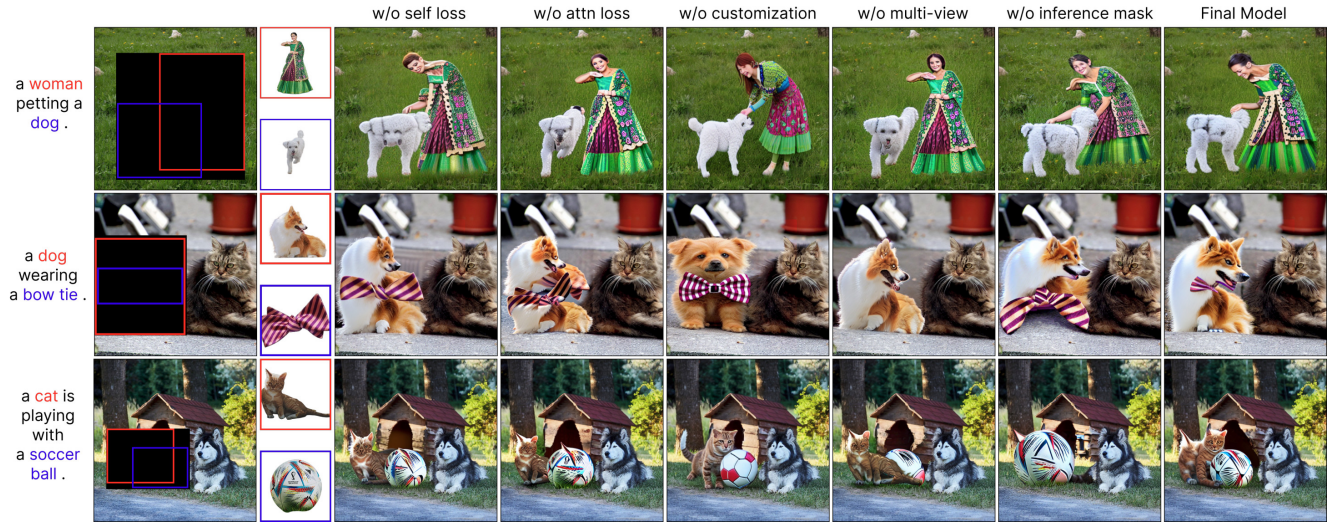

Figure 13. Visual examples for each ablation of the model. From left to right: (i) inputs (background, layout, objects and text), (ii) no self-attention loss, (iii) no self-attention or cross-attention loss, (iv) no joint training for compositing and customization, (v) no multi-view data (*i.e.* video data, manually collected data), (vi) no inference masking step, (vii) final model.

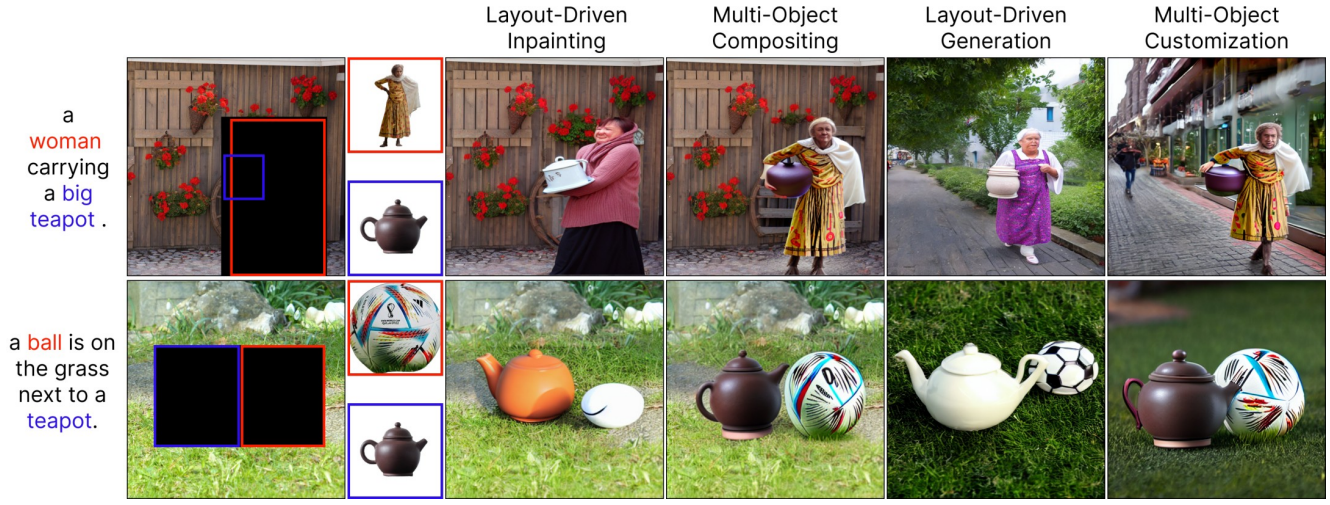

Figure 14. Visual examples for different applications of our model. Our model can operate on different modes such as: (i) layout-driven inpainting, (ii) multi-object compositing, (iii) layout-driven generation, (iv) multi-entity subject-driven generation.

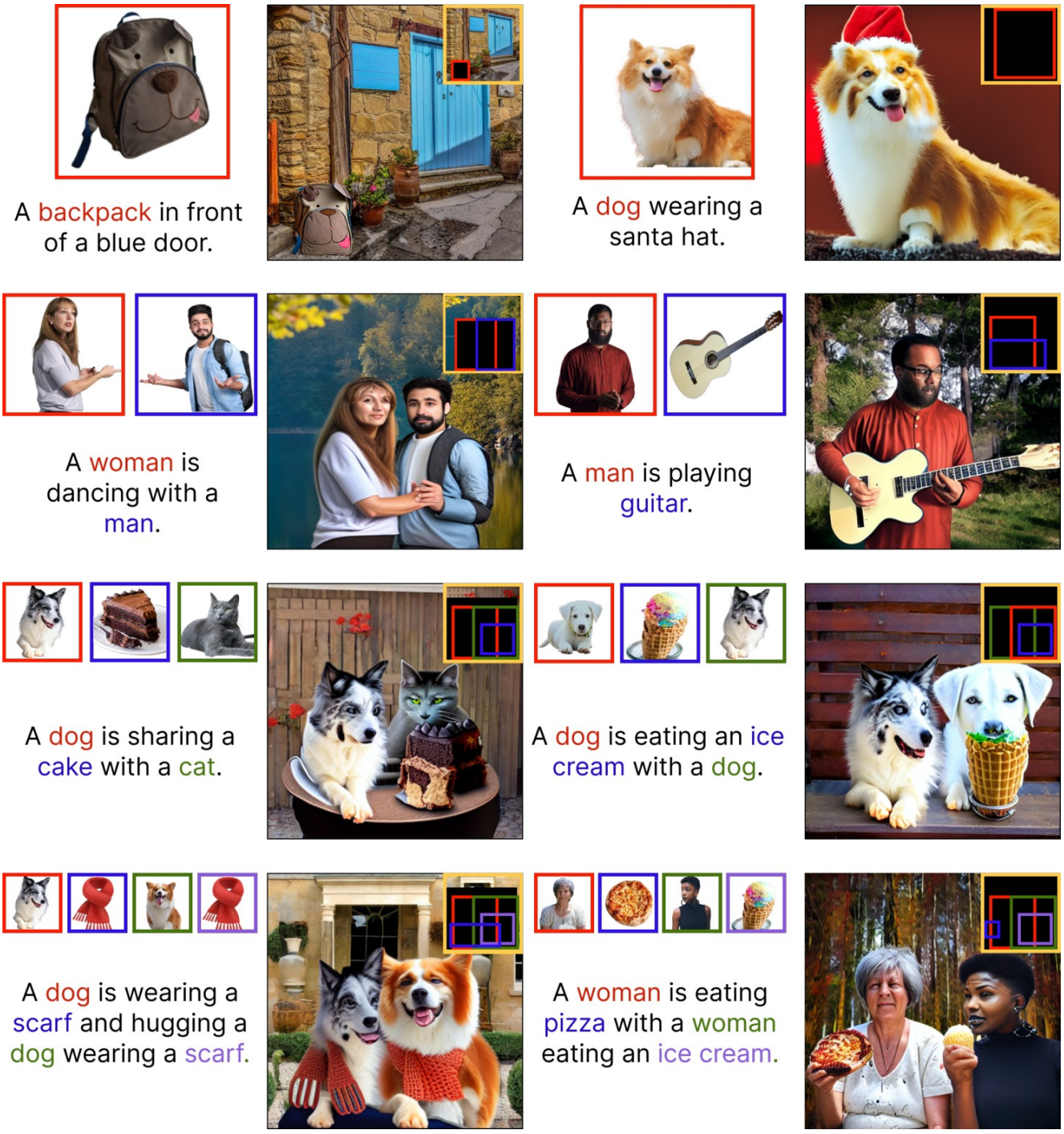

Figure 15. Visual Examples for Multi-Object Compositing (*left*) and Multi-Entity Subject-Driven Generation (*right*), using a variable number of grounding objects. *First Row:* One Object; *Second Row:* Two Objects; *Third Row:* Three Objects; *Forth Row:* Four Objects.

Figure 16. Visual Examples for Multi-Object Compositing in challenging scenarios, including: *(left)* harmonization, *(middle)* incomplete objects, and *(right)* imperfect segmentation and compositing of objects with different style (*i.e.* color, black and white).

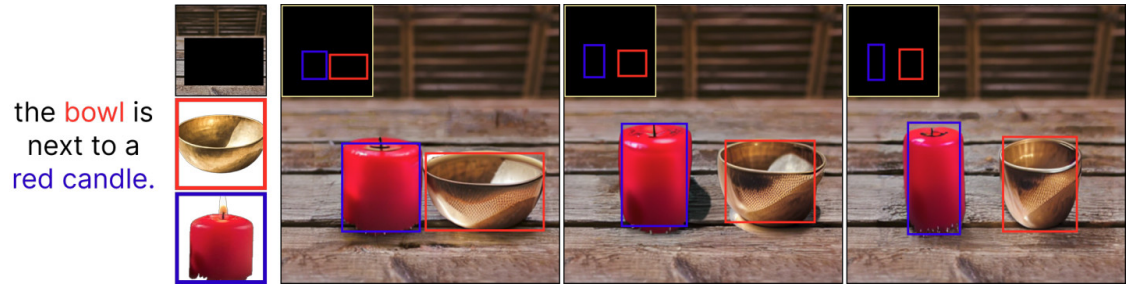

Figure 17. Visual Examples for Multi-Object Compositing via different layout inputs.

Figure 18. Visual Examples for Multi-Object Compositing using different grounded captions. The text caption can be used to edit poses and attributes of composited objects.

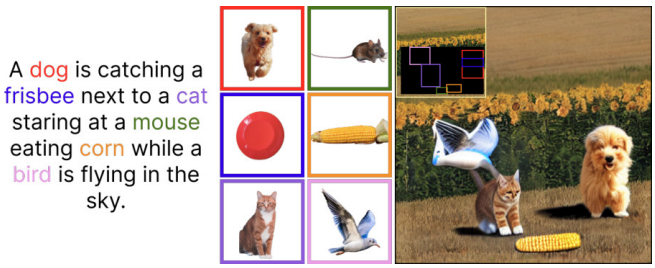

Figure 19. Visual example of a failed case from our model, depicting its limitation for compositing a high number of objects in a complex scenario.