Coherent Zero-Shot Visual Instruction Generation

Quynh Phung Songwei Ge Jia-Bin Huang
University of Maryland College Park
https://instruct-vis-zero.github.io

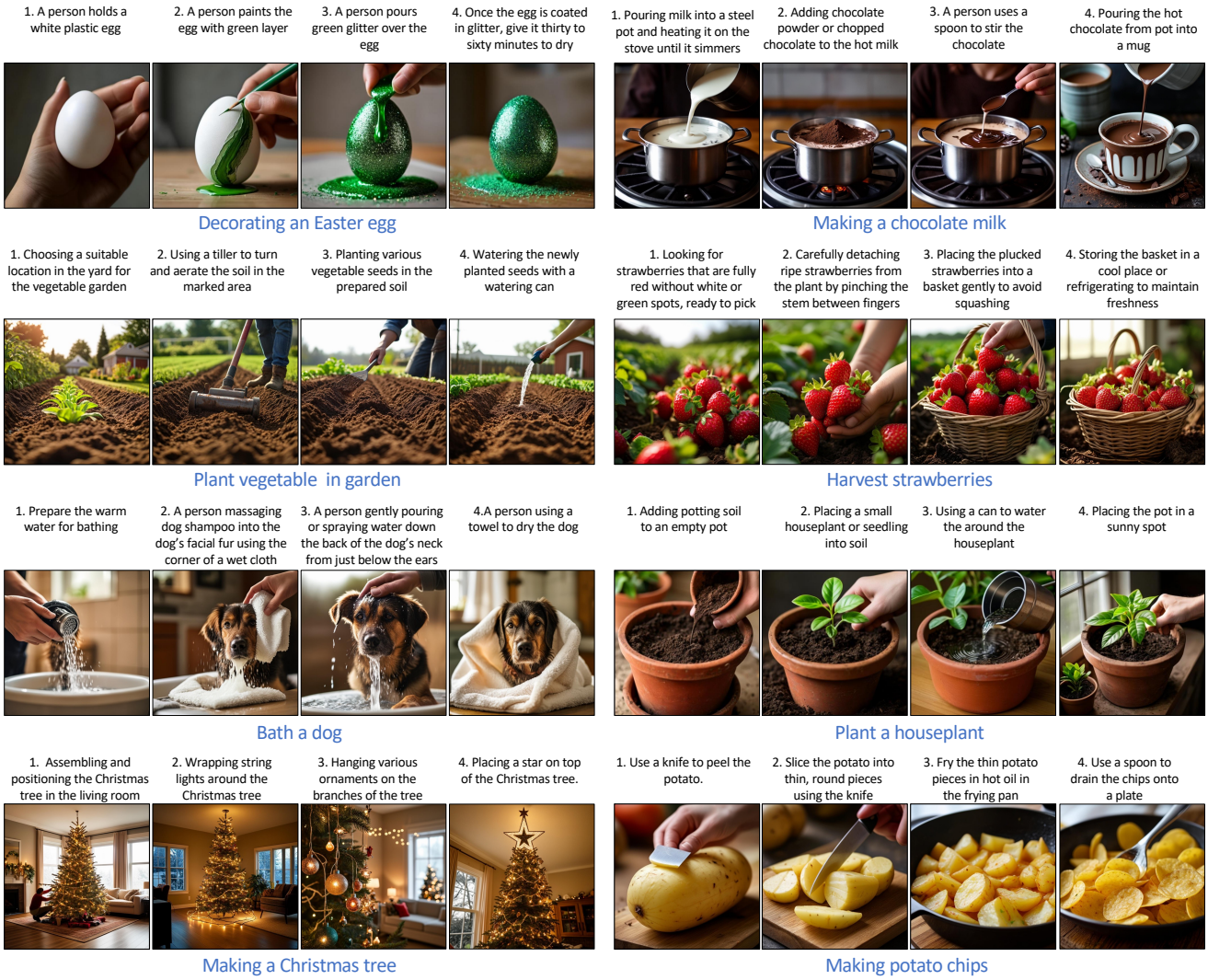

Figure 1. **Visual instruction generation results.** Given a sequence of textual instructions for a certain task, our method generates the *visual* instructions that illustrate the individual steps. Our method is training-free and thus preserves the quality and generalizability of the underlying image generation models. We showcase the generated visual instructions for different tasks from cooking to gardening. The samples possess high visual quality, align with the instructions, and maintain coherent object identity with desired changes at each step.

Abstract

Despite the advances in text-to-image synthesis, particularly with diffusion models, generating visual instructions that require consistent representation and smooth state transitions of objects across sequential steps remains a formidable challenge. This paper introduces a simple, training-free framework to tackle the issues, capitalizing

Figure 2. **Limitation of text-to-image generation in visual instructions task .** The crucial components of good visual instruction are 1) alignment with the text-based instruction and 2) coherence across different steps demonstrating the state changes. The current text-to-image generation methods focus only on the former. Consequently, the results may confuse the readers. In this paper, we develop a training-free method to enable a more coherent visual instruction generation.

on the advancements in diffusion models and large language models (LLMs). Our approach systematically integrates text comprehension and image generation to ensure visual instructions are visually appealing and maintain consistency and accuracy throughout the instruction sequence. We validate the effectiveness by testing multi-step instructions and comparing the text alignment and consistency with several baselines. Our experiments show that our approach can visualize coherent and visually pleasing instructions.

1. Introduction

Textual instructions are among the most prevalent tools for grasping new skills and knowledge and solving real-world problems, large and small. Generating visual illustrations of instructions has been a vital research problem [13, 28, 77] for their straightforwardness and ability to transcend language barriers, providing an intuitive understanding of the textual instructions [76]. In addition, apart from conveying information to humans, such visual data has also been widely adopted to train robotic policies [5, 14].

In this paper, we focus on generating *static* visual instructions. Unlike generating instructional videos [44, 80] that demand temporal consistency, generating static visual illustrations poses a relatively more approachable problem [67]. We leverage the recent advancements in text-to-image diffusion models [1, 25, 52], which has shown remarkable zero-shot capacity and photorealism, to visualize the instructions across a wide range of problem categories. Instead of fine-tuning the model on the instructional image dataset as in the existing methods [4, 43, 67], which could compromise the generation quality and limit itself to certain categories, we develop a training-free method of generating visual instructions.

Directly inputting the instructions to the text-to-image models incurs several issues, as shown by the examples in Figure 2. First, most instructions explain the procedure of actions, while the text-to-image model expects the description of the image content. As shown in step 2 of the "Making hot chocolate" example, the milk container becomes a glass bowl instead of the steel pot as mentioned in step 1 due to the lack of information in the instruction. This necessitates an approach to bridging the gap between instructional texts and the conditioning text used for image generation. Given the procedures at the current and previous steps, one needs to infer the proper image content at the current step. To this end, we propose an instruction re-captioning strategy [3, 69] to convert the instructional texts into actions and states using large language models (LLMs). We show that combining the action and state as the condition significantly enhances the quality and relevance of the generated illustrations.

In addition, the objects' identities may alter arbitrarily across different steps. For example, in the "planting a houseplan" example, the shape and texture of the pot in the first step are different from those in steps 3 and 4. This poses a common challenge when using text-to-image models to generate multiple images - there is often a lack of coherency across the generated illustrations. Recent studies have made progress in maintaining consistency for human portraits [10, 21, 29], where the identity features can be derived from the models trained on extensive human-centric datasets. However, our problem cannot benefit from the same idea since the identity features of the general object categories, such as kitchen utensils, are unavailable. Therefore, we refer to the general methods to achieve consistency, such as feature sharing and injection [71, 73].

Although these methods improve the generation consistency, they also induce the "over-consistent" problem in visual instruction generation. Specifically, many instructions involve changed objects among different steps. For instance, a recipe may contain a raw ingredient that is chopped, seasoned, and presented in entirely different states throughout the process. This inherent variability complicates the problem of maintaining object consistency and makes it unsuitable to use the vanilla feature-sharing strategy. To this end, we propose an adaptive feature-sharing method with finer-grained constraints. First, similar to the previous method, we adopt a local region constraint to en-

force sharing to happen on certain pixels. However, localizing the objects using attention maps becomes less reliable when the base model architecture differs from the UNet in Stable Diffusion [48]. Instead, we utilize large-scale segmentation models to produce the masks [30, 31, 34, 37, 40, 55]. Second, we apply a global constraint on the feature-sharing scale between each pair of steps based on their similarity. We exploit the reasoning capabilities of the LLMs to develop a similarity matrix that characterizes such state similarity. We show that our adaptive feature-sharing method enables the variability of objects across instructional steps while preserving object identity.

As showcased in Figure 1, our proposed method can generate high-quality, consistent visual representations based on instructional texts. We also perform quantitative experiments with various evaluation metrics to ablate individual components of our method on the text-image alignment and consistency. However, we notice that the image similarity metric only emphasizes the consistency aspect and conflicts to achieve variation across different steps. To this end, we propose a framework to evaluate the visual instruction generation quality using large-scale visual language models. We will release the code and full evaluation suite for reproducible research. The contributions of our work can be summarized as follows:

1. We develop a *training-free* method for generating visual instructions with pre-trained text-to-image diffusion models.
2. We propose an illustration re-captioning strategy, greatly improving the generation quality and relevance.
3. We introduce an adaptive feature-sharing method with finer-grained constraints to maintain object identity across different steps while allowing for necessary variations.
4. We present a framework to evaluate the visual instruction quality using large-scale visual language models. We show that our method can preserve the generation quality and show applicability across various categories.

2. Related work

Text-to-Image Generation with Diffusion Models . Diffusion models [24, 64–66] have become the ubiquitous choice for visual generation for their effectiveness in scaling on visual data distribution. When training on the large-scale datasets [6, 61], improved training and sampling with advanced techniques [23, 26, 45, 56], the state-of-the-art results have been achieved in image [1, 12, 50, 52, 54, 58] and video [2, 17, 20, 27, 63] generation. As learning from billions of data samples, the pretrained diffusion models have been shown to have great generalization capacity and thus been adapted to various downstream applications such as image editing [22, 42, 46], controllable generation [79], personalized image generation [35, 57], and even non-generative tasks [36, 68]. In this paper, we explore applying these pre-trained models to generate visual guidance in a zero-shot way.

Improving Consistency in Image Generation . Generating consistent images has been an important sub-problem in different tasks, including video generation [19, 33, 75, 81], multi-view image generation [41, 62, 72], and character generation [10, 21, 29, 71, 74]. Compared with these tasks, we focus on improving the consistency of the shared objects in different steps of the visual instructions. Unlike video or multi-view image generation, we don't enforce hard-constraint geometrical or temporal consistency. Different from the character generation that only cares about human identity, we need to deal with general objects that don't have ID feature extractors as those for humans [10, 21, 29, 74].

To achieve better consistency, many existing studies resort to fine-tuning [17, 29, 39] partially or completely the diffusion models on the consistent image data. In this paper, we tackle with zero-shot consistent image generation [71]. Previous studies have found that features in the diffusion models encode different information and can be utilized to control the generation. Cross-attention maps connect generation with text prompts and can be manipulated for additional textual information [1, 9, 22, 51]. Self-attention maps link pixels and encode rich structural information, which has been utilized to extract or modify the layouts [7, 47, 51]. The feature maps and noised latents contain more detailed information and can be used to reproduce the exact intact regions like background [18, 73]. In this paper, we build on these observations and propose several techniques to improve the consistency required in visual instruction generation. Specifically, we combine the controllability offered by these methods with the semantical understanding capacity of Large Language Models [15, 38] for finer-grained coherency.

Visual Instruction Generation . Both video and image can serve as the media for visual instructions. We focus only on generating static images as visual instructions [4, 43, 67]. Earlier works on instruction generation include recipe generation [11, 59, 60], while we are also interested in illustrating recipe steps with visual instructions. More recent works leverage the great generative power of pre-trained diffusion models and fine-tune these models on the visual instruction datasets [44, 78]. Bordalo et al. [4] integrates an Alpaca-7B model with a Stable Diffusion model for fine-tuning and generating sequences of visual illustration of recipes. GenHowTo [67] curates a dataset of states, actions, and resulting transformations triplets and trains a conditioned diffusion model on it. StackedDiffusion [43] fine-tuned a pre-trained text-to-image diffusion model with the stacked input on the Visual GoalStep Inference (VGSI) dataset [78].

Different from these existing methods, we aim to use the *pre-trained* diffusion model in a *zero-shot* manner for visual illustration generation.

3. Method

Given a set of instructions, we harness a pre-trained text-to-image diffusion model to generate the visual illustrations. As shown in Fig. 3, our approach contains two major stages. First, to fill the distribution gap between the instructions and image descriptions, we perform *in-context planning* with LLMs to re-caption the instructions. Second, given the re-captioned instructions, we propose an adaptive feature-sharing method for dynamic, coherent image generation. In both stages, we use off-the-shelf pre-trained models *without any extra training*.

3.1. Preliminaries

Text-to-image diffusion models. The text-to-image diffusion model incorporates a denoiser network D that is trained to estimate the noise in the current image, $\epsilon_t = D(\mathbf{x}_t; \mathbf{c})$, where t represents the timestep, and $\mathbf{c}$ denotes the conditional information embedding. During the inference time, an initial random Gaussian noise is iteratively denoised to generate a real image.

Self-attention layer is essential in D for integrating global information across the entire image. It redistributes the features from each spatial location to similar regions. Suppose that $\mathbf{x} \in \mathbb{R}^{w \times h \times d}$ denote the input feature map of some self-attention layer, where w, h, and d are the width, height, and dimension. Let $P = h \times w$ for simplicity. By applying linear mappings to the feature map $\mathbf{x}$ to obtain the *key* $K \in \mathbb{R}^{P \times d_k}$, *value* $V \in \mathbb{R}^{P \times d_v}$, and *query* $Q \in \mathbb{R}^{P \times d_k}$ where d_k is dimension of key K and query Q, d_v is dimension of value V, the self-attention map A^t at step t is generated by :

$$A^t = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) \in [0,1]^{P \times P}, \tag{1}$$

The attention maps mechanism calculates the similarity between query (Q) and key (K), which determines how much attention each value (V) receives.

KV sharing. To maintain object consistency across institutional steps, we draw inspiration from techniques used in video generation [32, 75]. These studies share keys and values within the self-attention layers across frames to allow the queries to attend to consistent elements in previous frames, implemented by concatenating keys and values:

$$K^+ = [K_1 \oplus K_2 \oplus \cdots \oplus K_N], V^+ = [V_1 \oplus V_2 \oplus \cdots \oplus V_N]$$

$$A_i^+ = \text{softmax}\left(\frac{Q_i K^{+T}}{\sqrt{d_k}}\right) \in [0,1]^{P \times N \cdot P} \tag{2}$$

$$H_i = A_i^+ \cdot V^+ \in \mathbb{R}^{P \times d_v}, \tag{3}$$

where$\oplus$ denotes concatenation, N is the number of images, and $i \in \{1, 2, \cdots N\}$. However, this technique may not generalize to our problem because the generated frames are supposed to be similar in both background and foreground, where visual instructions may contain dynamic elements.

Consistory [70] further proposed Self-Driven Self-Attention, focusing on sharing keys and values within individual objects across different frames. Specifically, it extracts object masks M from the cross-attention maps and assigns a $-\infty$ score to the self-attention maps for any pixel where the object mask value is zero, indicating that there should be no sharing in these regions. This updates the self-attention map calculation with the object masks:

$$M_i^+ = [M_1 \dots M_{i-1} \oplus M_i \oplus M_{i+1} \dots M_N], \tag{4}$$

$$A_i^+ = \text{softmax}\left(\frac{Q_i K^{+T}}{\sqrt{d_k}} + \log M_i^+\right) \in \mathbb{R}^{P \times N \cdot P}, \tag{5}$$

where $M_i = \mathbf{1}$, corresponding to the images i-th. This ensures that each image only attends to itself or object regions of other images. However, this method still assumes the objects to be fully consistent across different images, which is not often the case in visual instructions.

3.2. Re-captioning instructions as descriptive texts

Generating visual instructions from a sequence of textual instructions presents a significant challenge to current text-to-image models. To solve the problem, the model needs to understand objects' states and relationships across successive steps. For instance, as shown in the Fig. 2, consider the first two steps of making hot chocolate, where the milk is first poured into a steel pot, and the chocolate is then added to the milk. If one directly uses instructions as the input, the text-to-image models will not be informed with the context of "milk in the steel pot", leading to an incorrect container.

An immediate solution to this issue is to concatenate the adjacent instructions. However, this may lead to conflicting information as the instructions express different actions. (It can seen in the second row of Figure 9). Therefore, we propose leveraging LLMs' conversational understanding capabilities to re-caption the instruction into detailed input texts.

Since the text-to-image models take the descriptive image captions as the input, we prompt the LLM to predict the state of the scene given each instruction. The process is illustrated in that 3, where the states $S = (s_0, s_1, ...s_N)$ are predicted given a sequential set of N instructions $A = (a_0, a_1, ...a_N)$. We prompt the LLM by turns so that the model predicts the state of scene s_i given the current instruction and previous instructions and states. Therefore, the state s_i contains the scene context after the first i instructional steps, which we combine with the next instruc-

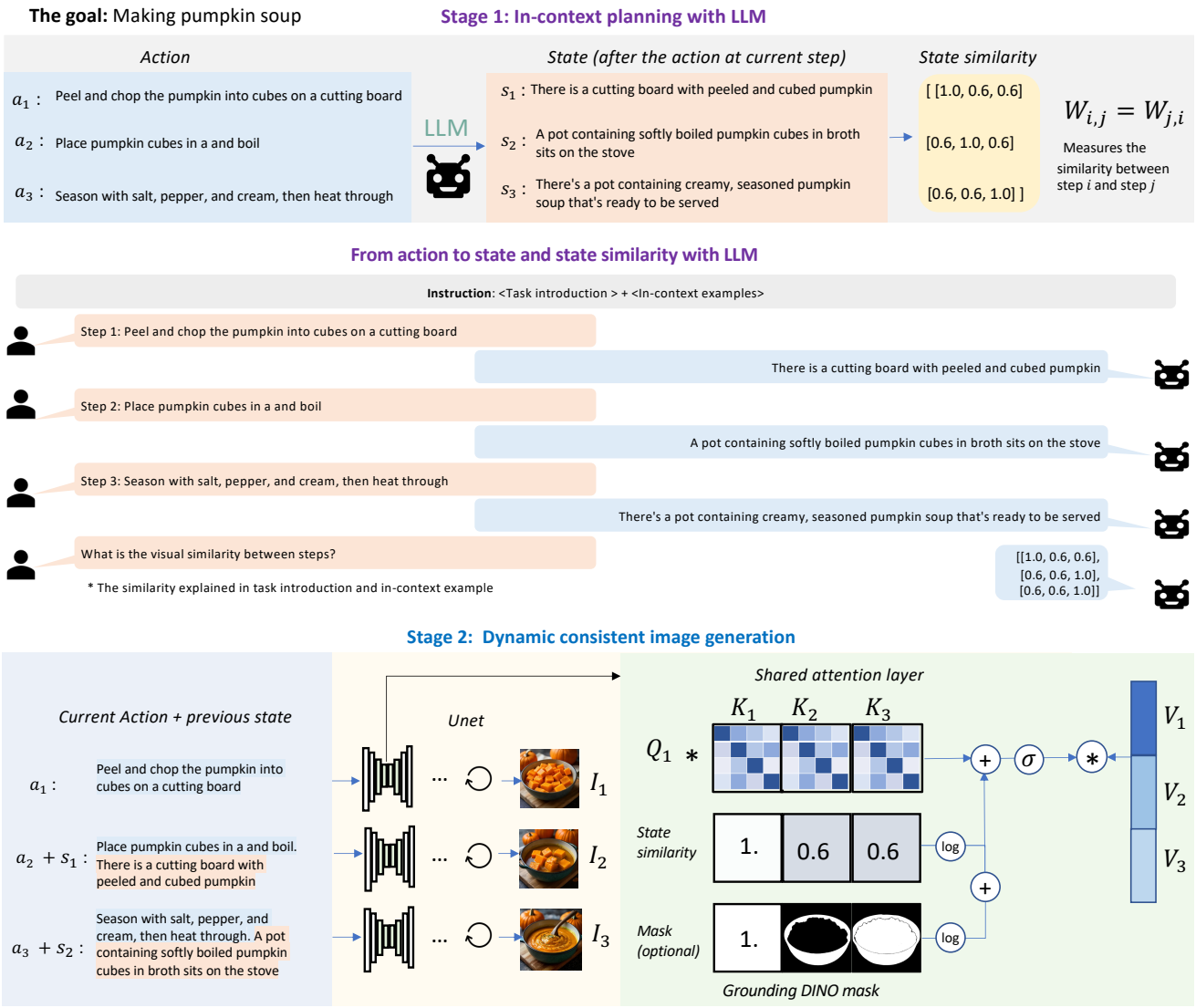

Figure 3. **Our framework for zero-shot instruction visualization.** Our framework operates in two distinct phases. In the first phase, we use an LLM (e.g., the GPT-4 model) to generate the scene state after each step in the list of instructions. The generated scene state helps guide the image generation in the next stage. We also ask the LLM to generate the similarity between states. This matrix, with each row indicating the visual similarity of a current visual step to others, guides the generation process. For example, to achieve high state similarity, we wish to maintain consistency as much as possible across the two steps. A low state similarity indicates the performed action changes the scene state substantially. In such cases, blindly encouraging *consistency* across steps may hurt the quality of the visualized instruction image. In the second phase, we utilize a shared attention layer—replacing the standard model—to allow queries from one image to access keys and values from others within the same instruction set. We enhance this sharing mechanism by applying standard attention masking, controlled by the similarity matrix, to finely tune the interaction between visual elements.

tion a_{i+1} as the input prompts to the text-to-image model:

$$p_i = \begin{cases} a_i + s_{i-1} & \text{if } i > 0 \\ a_i & \text{otherwise} \end{cases} \tag{6}$$

We find that this re-captioning method makes individual text prompts contain the necessary information that is revealed and should be maintained in the previous steps. As a result, the text-to-image model can achieve better continuity and context accuracy in the generated visual instruction sequence.

3.3. Dynamic consistent image generation

Our re-captioning technique ensures the description continuity in text prompts, while we still need to generate con-

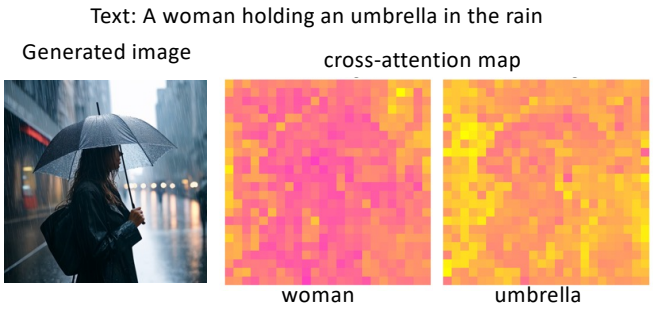

Figure 4. **Cross-attention map of Stable Cascade.** We visualize the corss-attention maps in stage C of Stable Cascade model. It is found that the attention maps are noisy and fail to accurately delineate the specific regions of the main objects: woman and umbrella.

sistent content shared across instructions. Unlike previous methods like video or story generation, which require absolute consistency on the entire scene or the object region, instructional generation requires a more nuanced consistency. As discussed in the previous sections, some visual instruction steps may demand maintaining object consistency, while others may involve significant transformations of the objects or background according to the procedure. To this end, we propose an adaptive KV-sharing method based on local region and state similarity constraints.

KV sharing with local region constraint. Similar to Consistory [70], visual instruction generation only cares about consistency within certain regions. We identify these consistent objects in the instructions during our in-content planning phase. With our re-captioned instructions, the images generated realize most of the information about the instructions, like the layout, except for the object consistency. Therefore, we propose to use off-the-shelf segmentation models to generate the object masks [40, 55]. We also explored using cross-attention maps to produce the mask, as in Consistory. However, the link between text and image can be weak and noisy in the state-of-the-art text-to-image models [49] as shown in Figure 4. Instead, we show that existing segmentation models work well with generated images and are not overfitted to a single model architecture.

KV sharing with state similarity constraint. To manage the dynamic scenario of generating consistent objects with variations, we continuously generalize the KV sharing. Instead of only sharing or not sharing KV features among different regions, we regulate the scaling of sharing with a state similarity matrix. This matrix controls the degree of similarity between each pair of instructional steps. By adjusting its values, one can tailor the visual output to highlight consistency or emphasize transformation, depending on the instructions. To automatically infer this matrix from the instructions, we provide thoughtful instruction and in-context examples to LLMs. Let $W \in [0,1]^{N\times N}$ be the state similarity matrix, where N is the number of steps. $W_{i,j}$ represents the overall similarity between the i-th and the j-th steps.

We inflate $W_{i,j}$ into a matrix S_j, which is used for computing the attention matrix of the i-th image. We use the matrices derived from the similarity matrix to regularize the attention maps in Equation 5 is as follows:

$$S_i^+ = [S_1 \ldots, S_{i-1} \oplus 1 \oplus S_{i+1}, \ldots, S_N] \tag{7}$$

$$A_i^+ = \text{softmax}\left(\frac{Q_i K^{+T}}{\sqrt{d_k}} + \log(S_i^+) + \log(M_i^+)\right) \in \mathbb{R}^{P\times N\cdot P} \tag{8}$$

Here, A_i^+ represents the adjusted attention weights, where the flow of information in self-attention is scaled inversely by the values in S_i^+. When S_j approaches zeros (indicating no similarity between steps i-th and j-th states), the sharing of information between those specific steps is nullified. The magnitude of information sharing between the i-th and j-th steps is substantial when the values of S_j are high, encouraging more consistent regions to be generated. This method allows us to precisely control the trade-off between consistency and variation across different steps in the visual instruction generation process, ensuring that each generated step is appropriately aligned with the instruction and the other images in the sequence.

4. Experiments

4.1. Experiment setup

In this section, we evaluate our method for generating visual illustrations of textual instructions. We propose a framework to leverage the vision language models (VLMs) for the evaluation. We also perform ablation studies on our individual designs.

Evaluation Following the previous studies, we utilize CLIP-Score [53] to measure the text-image alignment, and Dreamsim [16] and L2_Dinov2 [8] to evaluate the consistency. However, the tasks in the instructions span a wide range of categories. Each of these may have a distinct goal beyond simple object consistency. Understanding the quality of the visual illustration demands nuanced reasoning—for instance. At the same time, some scenarios necessitate maintaining object consistency, and others require deliberate changes in the object's state, such as cutting or decorating. Therefore, conventional metrics often do not evaluate such a complex task. Instead, we turn to recent VLMs like GPT-4V and Gemini-Pro 1.5, which show great visual understanding and reasoning capacity. We mainly evaluate in the following aspects:

1. **Textual Alignment** measures how well the visual content matches the textual instructions.
2. **Continuity** evaluates the transition process in a sequence of images or within elements of a single image.
3. **Consistency** assesses whether the objects in the image remain the same throughout a sequence or within the context of the image.
4. **Relevance** determines how focused the image is on the main object or theme as described in the input.

To apply these metrics, we use carefully designed instructions and in-context examples to query vision-language models. We provide a pair of generated images, one by our method and the other by the baseline method, each time. The VLM is requested to pick the image better in the above aspects.

Dataset To facilitate the study of visualizing textual instructions, we use GPT-4 to generate 200 goals and instructions for wide-ranging tasks, including cooking, gardening, and decorating. Each instruction contains 3 to 5 steps.

Implementation details. Our method includes two main components. We use GPT-4 API for in-context planning. We provide all the in-context examples and prompts in the appendix. However, we note that users can achieve similar quality using ChatGPT3 or 4. For dynamic, consistent image generation, we use Stable Cascade [49] as our text-to-image model, which has three stages: A, B, and C. Based on the report and exploration, we find that only stage C forms the image based on the text condition (stage B uses text as a condition, but in this stage, the text condition barely affects the generation results). Thus, we apply our adaptive KV sharing method in stage C. Specifically, we apply it in the first 15 steps, which total 20.

4.2. Quantitative results

We evaluate our method and baseline approaches using vision-language models and show results in Figure 5 and Figure 6. We quantitatively compare using different conditioning texts as input in Figure 5, demonstrating that our re-captioning approach achieves overall better results than baseline methods relying solely on image generation instructions. Our method shows consistently improved behavior across all four aspects. We then quantitatively compare different feature-sharing methods in Figure 6. Our adaptive KV sharing methods improve almost all metrics using Gemini and GPT-4V.

We also show the quantitative results with traditional metrics in Table 1 and 2. which demonstrates that our method is comparable with baselines. Specifically, as shown in Table 2, applying adaptive KV sharing leads to less identical results as desired while greatly improving the text alignment.

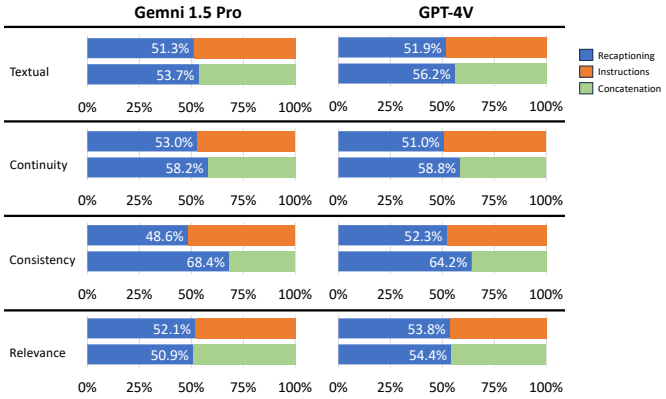

Figure 5. Evaluation of different design choices of text prompts using LLM, including Gemini and GPT-4. Among different evaluation aspects, including text alignment, continuity, consistency, and relevance, our choice of concatenating action and state beats using action only or concatenating with previous actions.

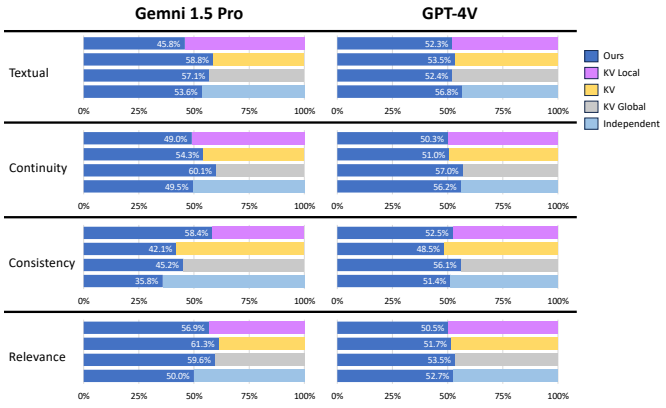

Figure 6. Evaluation of different consistency methods using LLM, including Gemini and GPT-4. Using both masks and weights achieves the best performances among all the choices overall.

Table 1. Ablation study of different design choices of the text prompts. Using instructions only provides the best text image alignment while concatenating with previous instructions or states improves the coherency.

Input texts	Clip-score ↑	Dreamsim↓	L2-Dinov2↓
Instructions	**0.6980**	0.0.4829	47.7013
Concatenation	0.4559	**0.3433**	**38.8717**
Re-captioning	0.5138	0.3797	41.6487

4.3. Qualitative results

Figure 9 illustrates the impact of various design choices for prompts on the coherence and alignment of the generated instructions. When only the action from a single step is used as input, the generated images lack coherence due to unawareness of the context. For instance, in "cooking Hot

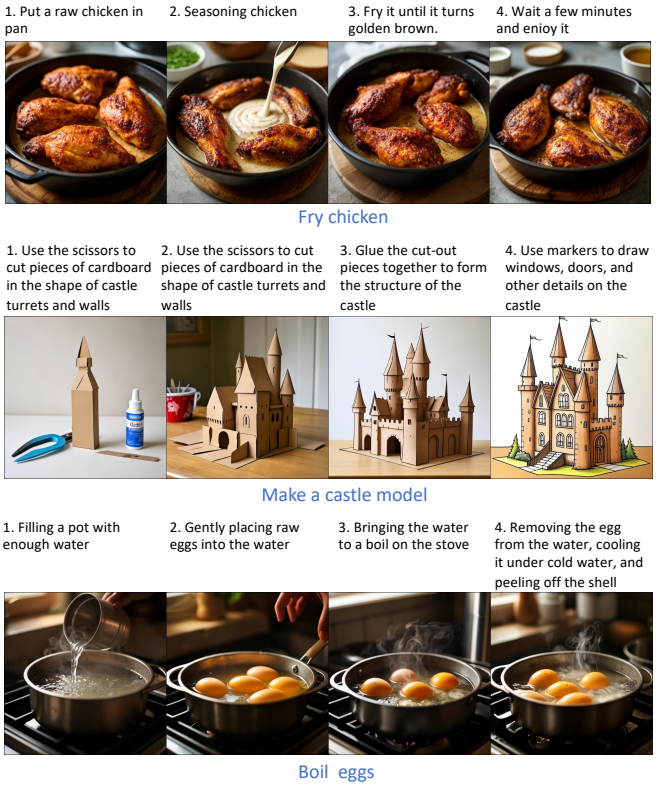

Figure 7. **Failures cases.** Due to the limited capability of the text-to-image model, it cannot generate raw chicken or raw eggs.

Table 2. Quantitative evaluation of our methods to achieve object coherency. Our method, which uses both local region mask and global similarity constraints, greatly improves the text alignment for its flexibility in realizing the variation of object states across steps, which is shown as an increase in similarity measurement.

Global	Local	Clip-score ↑	Dreamsim↓	L2-Dinov2 ↓
×	×	0.4929	**0.3638**	**40.5037**
✓	×	0.5358	0.3792	41.5642
×	✓	0.5138	0.3799	41.6487
✓	✓	**0.6708**	0.4377	45.6434

Chocolate", the pot is only mentioned in step 1. As a result, steps 2 and 3 do not know that the pot is in the stove and contains the hot milk. Therefore, the model fails to maintain the context; the pot is not in the stove and does not contain the milk anymore in steps 2 and 3. In scenarios where current and previous actions are concatenated, the model can better understand the context. However, such models tend to prioritize the most recent action at the cost of accurately generating earlier ones. Concatenating action and state significantly enhances context understanding, leading to more accurate image generation based on prior steps. For instance, this method ensures the generated image accurately

Figure 8. **Visual comparisons of consistency across steps.** Here, we use the proposed prompting approach and focus on validating the various ways of sharing keys and values in attention layers for consistent image generation. All methods take the same input text prompt. **KV**: sharing across early steps; **KV local**: sharing controlled by masks; **KV global**: sharing controlled by the proposed state similarity matrix; **Ours**: sharing controlled by both masks and the state similarity matrix. For example, in the omelet, naïvely sharing the key and value across steps cause *content leaking* (e.g., the pan in steps 3 and 4 looks like the bowl in steps 1 and 2). With weight control, the method can maintain consistency, avoiding leaking feature while respecting the action. Adding mask control helps improve textual alignment. (e.g, the actions are more align with text)

places the pot on the stove in the case of a pot and cooking. In another example involving cookies, the approach allows for generating images of cookies with chocolate chips, ensuring the action aligns closely with the input text.

In Figure 8, the stable cascade model, when generating images independently, fails to ensure consistency across different steps. Using basic KV sharing makes images in different steps globally similar. However, errors occur, such as a pan being misinterpreted as a bowl and the actions not aligning with the input text well. Applying a local region

Figure 9. **Visual comparisons with concatenating steps. Ours**: concatenating current instruction and inferred states; **Concatenation**: concatenating current and previous instructions; **Instruction**: using only current instruction

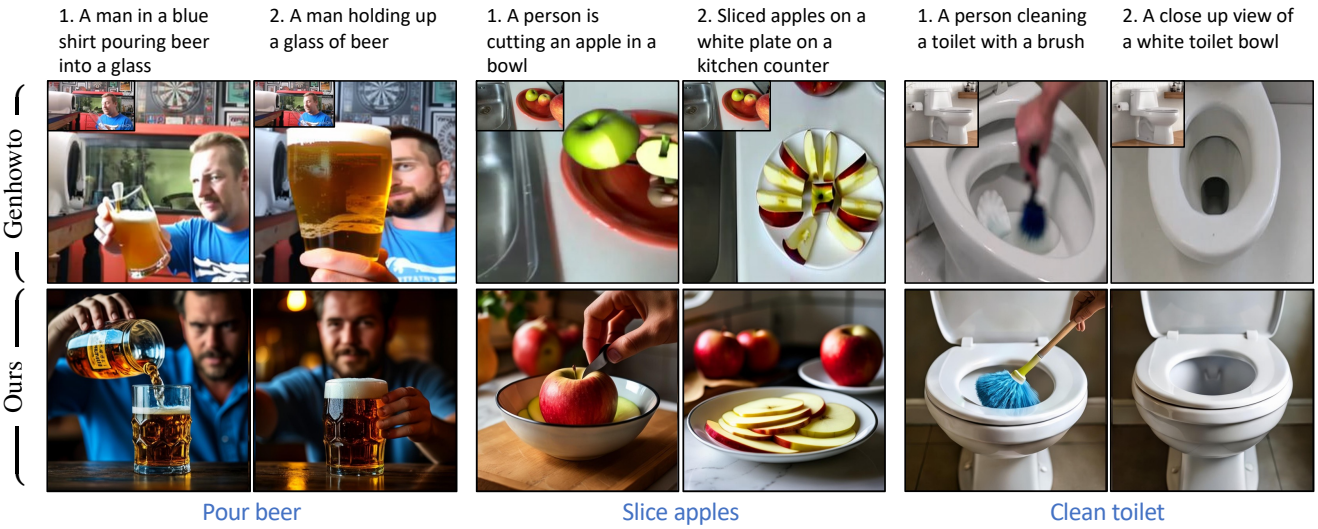

Figure 10. **Visual comparisons with Genhowto [67]**. Comparison of our method with the GenHowTo method on instructions from their paper. The small images in the top left of the GenHowTo images are the real images used as inputs for their method. Our free-training method is compatible with their pretrained model and produces visually more pleasing results without requiring a real image as input.

mask addresses some issues of action omission, though it still misrepresents a bowl as a pan. Regularizing KV sharing with state similarity resolves the problem of missing attributes between steps, yet it inadequately captures the action, as seen in the poor depiction of pouring eggs. Ours combines both methods and effectively solves the issues of missing actions and object misrepresentation across different steps.

In Figure 10, we compare our method with GenHowTo [67], a pretrained model for generating actions and states from real images. We use prompts from the GenHowTo test dataset, which belongs to the same categories as their training data. Note that our method is training-free and does not require a real input image as input. Ours produces comparable results regarding textual alignment, consistency, and even better visual quality. This is because our

approach leverages a state-of-the-art text-to-image model. Unlike GenHowTo [67], which requires re-training the model for each new base text-to-image model and may compromise the generation quality due to the training data quality.

5. Failure cases and discussion

We shown several failure cases within our model in Figure 7, where it did not accurately generate the stated objects and attributes from the instructions, thereby misleading the process. For example, in a step described as frying raw chicken, the model erroneously generated an image of cooked chicken. Similarly, in another instance involving boiling a raw egg, the output also deviated from the specified raw state. Additionally, the model exhibited a bias towards rendering castles in a painting style, which led to inconsistencies in the style of generation across different tasks. We believe that as text-to-image models improve in the future, our method will greatly benefit from reduced limitations inherent in current text-to-image models.

6. Conclusion

In this paper, we tackle the problem of generating static visual illustrations from textual instructions by leveraging pretrained diffusion models, enabling high-quality generation without expensive fine-tuning. We propose a framework to address the unique challenges of maintaining object consistency across instructional steps while managing the variability of objects that change across states. Our extensive evaluations demonstrated that our method outperforms baseline models in both consistency and accuracy.

References

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

[2] Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Yuanzhen Li, Tomer Michaeli, et al. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*, 2024.

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[4] João Bordalo, Vasco Ramos, Rodrigo Valério, Diogo Glória-Silva, Yonatan Bitton, Michal Yarom, Idan Szpektor, and Joao Magalhaes. Generating coherent sequences of visual illustrations for real-world manual tasks. *arXiv preprint arXiv:2405.10122*, 2024.

[5] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[6] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset`, 2022.

[7] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2023.

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

[9] 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.

[10] Weifeng Chen, Jiacheng Zhang, Jie Wu, Hefeng Wu, Xuefeng Xiao, and Liang Lin. Id-aligner: Enhancing identity-preserving text-to-image generation with reward feedback learning, 2024.

[11] Prateek Chhikara, Dhiraj Chaurasia, Yifan Jiang, Omkar Masur, and Filip Ilievski. Fire: Food image to recipe generation. 2024.

[12] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023.

[13] Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[14] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. In *International Conference on Machine Learning (ICML)*, 2021.

[15] Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[16] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023.

[17] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-

Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[18] Songwei Ge, Taesung Park, Jun-Yan Zhu, and Jia-Bin Huang. Expressive text-to-image generation with rich text. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[19] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. In *International Conference on Learning Representations (ICLR)*, 2024.

[20] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023.

[21] Zinan Guo, Yanze Wu, Zhuowei Chen, Lang Chen, and Qian He. Pulid: Pure and lightning id customization via contrastive alignment. *arXiv preprint arXiv:2404.16022*, 2024.

[22] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *International Conference on Learning Representations (ICLR)*, 2023.

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. 2020.

[25] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[26] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 23:47–1, 2022.

[27] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[28] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

[29] Jiehui Huang, Xiao Dong, Wenhui Song, Hanhui Li, Jun Zhou, Yuhao Cheng, Shutao Liao, Long Chen, Yiqiang Yan, Shengcai Liao, et al. Consistentid: Portrait generation with multimodal fine-grained identity preserving. *arXiv preprint arXiv:2404.16771*, 2024.

[30] Chuong Huynh, Yuqian Zhou, Zhe Lin, Connelly Barnes, Eli Shechtman, Sohrab Amirghodsi, and Abhinav Shrivastava. Simpson: Simplifying photo cleanup with single-click distracting object segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[31] Chuong Huynh, Seoung Wug Oh, Abhinav Shrivastava, and Joon-Young Lee. Maggie: Masked guided gradual human instance matting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[32] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023.

[33] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[34] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023.

[35] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[36] Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[37] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023.

[38] Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024. Featured Certification.

[39] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[40] 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.

[41] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. In *International Conference on Learning Representations (ICLR)*, 2024.

[42] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2022.

[43] Sachit Menon, Ishan Misra, and Rohit Girdhar. Generating illustrated instructions. *arXiv preprint arXiv:2312*, 2023.

[44] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic.

Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[45] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning (ICML)*, 2021.

[46] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. 2023.

[47] 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 *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[48] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023.

[49] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023.

[50] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

[51] Quynh Phung, Songwei Ge, and Jia-Bin Huang. Grounded text-to-image synthesis with attention refocusing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[52] 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.

[53] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

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

[55] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.

[56] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[57] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[58] 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. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[59] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. Learning cross-modal embeddings for cooking recipes and food images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[60] Amaia Salvador, Michal Drozdzal, Xavier Giró-i Nieto, and Adriana Romero. Inverse cooking: Recipe generation from food images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[61] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[62] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. MVDream: Multi-view diffusion for 3d generation. In *International Conference on Learning Representations (ICLR)*, 2024.

[63] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. In *International Conference on Learning Representations (ICLR)*, 2023.

[64] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021.

[65] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. 2019.

[66] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

[67] Tomáš Souček, Dima Damen, Michael Wray, Ivan Laptev, and Josef Sivic. Genhowto: Learning to generate actions and state transformations from instructional videos. *arXiv preprint arXiv:2312.07322*, 2023.

[68] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[69] Omost Team. Omost github page, 2024.

[70] Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. Training-free consistent text-to-image generation. *arXiv preprint arXiv:2402.03286*, 2024.

[71] Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. Training-free consistent text-to-image generation. 2024.

[72] Hung-Yu Tseng, Qinbo Li, Changil Kim, Suhib Alsisan, Jia-Bin Huang, and Johannes Kopf. Consistent view synthesis with pose-guided diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[73] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[74] Qinghe Wang, Baolu Li, Xiaomin Li, Bing Cao, Liqian Ma, Huchuan Lu, and Xu Jia. Characterfactory: Sampling consistent characters with gans for diffusion models. *arXiv preprint arXiv:2404.15677*, 2024.

[75] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[76] Aman Yadav, Michael M Phillips, Mary A Lundeberg, Matthew J Koehler, Katherine Hilden, and Kathryn H Dirkin. If a picture is worth a thousand words is video worth a million? differences in affective and cognitive processing of video and text cases. *Journal of Computing in Higher Education*, 23:15–37, 2011.

[77] Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations (ICLR)*, 2024.

[78] Yue Yang, Artemis Panagopoulou, Qing Lyu, Li Zhang, Mark Yatskar, and Chris Callison-Burch. Visual goal-step inference using wikihow. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

[79] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[80] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI Conference on Artificial Intelligence*, 2018.

[81] Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. Storydiffusion: Consistent self-attention for long-range image and video generation. 2024.

A. Appendix

A.1. Additional discussion and details about large language models

Full prompt for GPT-4 Our complete prompt for GPT-4 includes three main components:

- **Instruction:** This specifies the task and defines the output format, helping GPT-4 perform effectively in layout generation tasks.
- **In-context exemplars:** These enhance the model's capability for the task by providing multiple examples. These examples help the model understand the context better and produce the desired bounding boxes and corresponding labels.
- **User prompt:** This is appended to the instruction and supporting examples. The model then continues the conversation based on the user prompt and provides the layout in the specified format.

When users provide a prompt (user prompt), it is combined with the predefined text to create a complete prompt as shown in Table 3. The GPT-4 API then processes this complete prompt and returns the information about each steps, similarity matrix and the main objects in each step.

Metrices using Multimodel Gemini and GPT-4V We evaluate our genererated visual instruction using Gemini15-pro and gpt-4V to evaluate four aspects, we use the instruction sjown in Figure 12 to guide the multi-model to assess our visual instructions. We shuffle the order of the two methods which are compared to avoid the multi-model bias toward the order.

Table 3. The full prompt for gpt4 api to generate instructions.

Role	Content
Instruction	System: ”You are ChatGPT-4, act like visual and instructional experts, generate step-by-step how to do something. each step include the action to indicate how people interact with objecs, and state to show state of objects after finish this action. And relation matrix is the correlation of one step with others in visual. object field indicate the objects in each step similar with privious step in some extends: similar(total similar), shape similar(only similar shape), texture similar(transform shape, only same texture)”
In-context examples	User: ”The instruction on decorating a cake in 2 steps.” Assistant: [examplers in Figure 11]
User prompt	User : ”The instruction on” + [user prompt]

```
{
    "goal": "Decorating a Cake",
    "steps": [
        {
            "step": "Setting the Cake
               on a Platter",
            "object": [["cake", "new"],
                ["platter", "new"]],
            "action": "Set the baked
               cake on a platter.",
            "state_of_main_object": "A
               baked cake on the
               platter."
        },
        {
            "step": "Applying Icing",
            "object": [["cake", "
               similar shape", 1], ["
               spoon", "new"]],
            "action": "Person using a
               spoon to place some
               icing on the top of the
               cake.",
            "state_of_main_object": "
               The cake covered by
               icing."
        }
    ],
    "relation": [
        [1.0 , 0.5, 0.4, 0.3],
        [0.9, 1.0 , 0.5, 0.4],
        [0.8, 0.9, 1.0, 0.4],
        [0.7, 0.8, 0.9, 1.0 ]
    ]
}
```

Figure 11. In-context examplers for full prompts

```
Our task here is to compare visual step-by-step instructions, generated from the
    same step-by-step textual instruction. We want to decide which one is better
    according to the provided criteria.
# Instruction
1. Text prompt and Asset Alignment: Focus on whether the key elements mentioned in
     the text are clearly visible and identifiable in the image. The visual is good
     if all key elements are clearly depicted and easily identifiable.
2. Continuity: This measures how well the image captures the progression from the
    previous step(s), maintaining context and demonstrating the changes or actions
    described in the current step. The visual is good if the image effectively
    shows the progression from previous steps and integrates new elements/actions
    as described in the current step.
3. Consistency: Evaluates whether the same objects are used consistently across
    all images in a way that reflects their continued presence and role as
    described in the text. This is particularly important for objects that are
    central to the action or instructions. For example, a pot in first step should
    look like the pot mentioned other step, even it can be in different views.
4. Relevance: Assesses whether the visual focuses on the most critical aspect of
    the step as described in the text. The visual is good if the visual focuses
    precisely on the primary action or element described in the step.
Take a really close look at each of the multi-image instructions for the
    corresponding textual instruction before providing your answer.
When evaluating these aspects, focus on one of them at a time.
Try to make independent decisions between these criteria.
# Output format
To provide an answer, please provide a short analysis for each of the
    abovementioned evaluation criteria. The analysis should be very concise and
    accurate.
For each of the criteria, you need to make a decision using these options:
1. The first row visual is better;
2. The second row visual is better;
... or Cannot decide.
IMPORTANT: PLEASE USE THE 'Cannot decide' OPTION SPARSELY.
Then, in the last row, summarize your final decision by <option for criterion 1> <
    option for criterion 2> <option for criterion 3> <option for criterion 4>.
# Example

Analysis:
1. Text prompt and Asset Alignment: The first one ...; The second one ...; The
    first/second/third/... one is better or cannot decide.
2. Continuity: The first one ...; The second one ...; The first/second/third/...
    one is better or cannot decide.
3. Consistency: The first one ...; The second one ...; The first/second/third/...
    one is better or cannot decide.
4. Relevance: The first one ...; The second one ...; The first/second/third/...
    one is better or cannot decide.
Final answer:
x, x, x ,x (e.g., 1, Cannot decide, 3, 1/ 2, Cannot decide,5, 1 / 1, 3, 2,4)
```

Figure 12. Instruction for Gemini and GPT-4V to asses the visual instruction