A EXTENDED LITERATURE REVIEW

As aforementioned in the main text, in this section, we provide an additional review of related works, on (1) Retrieval-augmented Generative Models; and (2) Text-guided Image Editing.

Retrieval-Augmented Generative Models Knowledge grounding has also drawn significant attention in the natural language processing (NLP) community. Different semi-parametric models like KNN-LM (Khandelwal et al., 2019), RAG (Lewis et al., 2020), REALM (Guu et al., 2020), RETRO (Borgeaud et al., 2021) have been proposed to leverage external textual knowledge into the transformer language models. These models have demonstrated great advantages in increasing the language model's faithfulness and reducing the computation/memory cost. Such attempts have also been made in visual tasks like image recognition (Long et al., 2022), 2-D scene reconstruction (Siddiqui et al., 2021), and image inpainting (Xu et al., 2021). Our proposed method follows the same theme to incorporate visual knowledge into a pre-trained text-to-image generation model to help the model generalize to long-tail entities or even unseen entities without scaling up the parameters.

Text-Guided Image Editing The work of text-guided image editing aims at preserving the object's appearance while changing certain contexts in the image. Previously, GANs (Goodfellow et al., 2014) have been used to achieve significant performance on image editing (Zhu et al., 2016; Abdal et al., 2019; Zhu et al., 2020; Roich et al., 2021; Tov et al., 2021; Wang et al., 2022; Alaluf et al., 2022). The problem is also known as inversion as it normally requires finding the initial noise vector added in the generation process. More recently, Prompt-to-Prompt (Hertz et al., 2022) propose to use pre-trained text-image models for image editing. Image editing is focused on performing in-place modifications to the input image, either changing the global styles or editing a local region specifically without modifying the object's appearance. However, we treat the retrieved image as 'knowledge' and ground on it to synthesize new images. Thus, we are not restricted to in-place modifications and are able to perform more sophisticated transformations over the objects.

B IMAGEN EXAMPLES ON FREQUENT ENTITIES

Here we demonstrate some images being generated by Imagen Saharia et al. (2022) regarding frequent entities and display them in Figure 8.

Figure 8: Imagen is excellent in generating images containing highly frequent entities.

C WIKIIMAGES DATASET

The WikiImages dataset is taken from WebQA (Chang et al., 2022). Images were crawled from Wikimedia Commons via the Bing Visual Search API. Since lots of Wikimedia's topics are not visually interesting, the authors seeded with natural scenes and gradually refine the search pool to obtain more interesting images. The images are mostly containing entities from Wikipedia or WikiData. However, the original dataset still contains heavy noises. Therefore, we apply further filtering to obtain the more plausible ones for image generation. Specifically, we remove all the image-text pairs with text lengths larger than 15 tokens and all the text with a date or wiki-id information. More examples from WikiImages are displayed in Figure 9.

Figure 9: More examples from the WikiImages dataset.

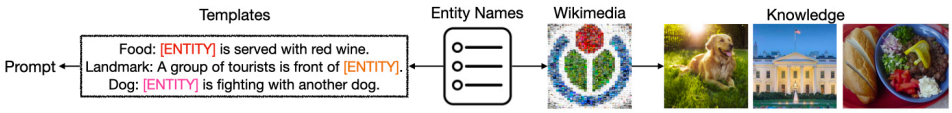

Figure 10: The construction process of EntityDrawBench. We first list entity names and then find their source images from Wikimedia, and finally generate prompts related to these entities.

D ENTITYDRAWBENCH

For dog breeds, we sample 50 from Wikipedia Commons[3] as our candidates. For landmarks, we sample 50 from Google Landmarks (Weyand et al., 2020) as our candidate. For foods, we sample 50 from Wikipedia[4] as our candidates. We use appropriately paired source images as the retrieved 'knowledge'. For each entity category, we write 5 prompt templates with an entity name placeholder, which describes the entity in different scenes. Each entity will sample a template and replace the placeholder with the entity's name to generate a prompt.

We list all the prompt templates as follows:

- '[DOG] is sleeping on the ground.',
- '[DOG] is running by the river.',
- '[DOG] is catching a frisbee.',
- '[DOG] is taking a shower.',

[3] https://commons.wikimedia.org/wiki/List_of_dog_breeds
[4] https://en.wikipedia.org/wiki/List_of_cuisines

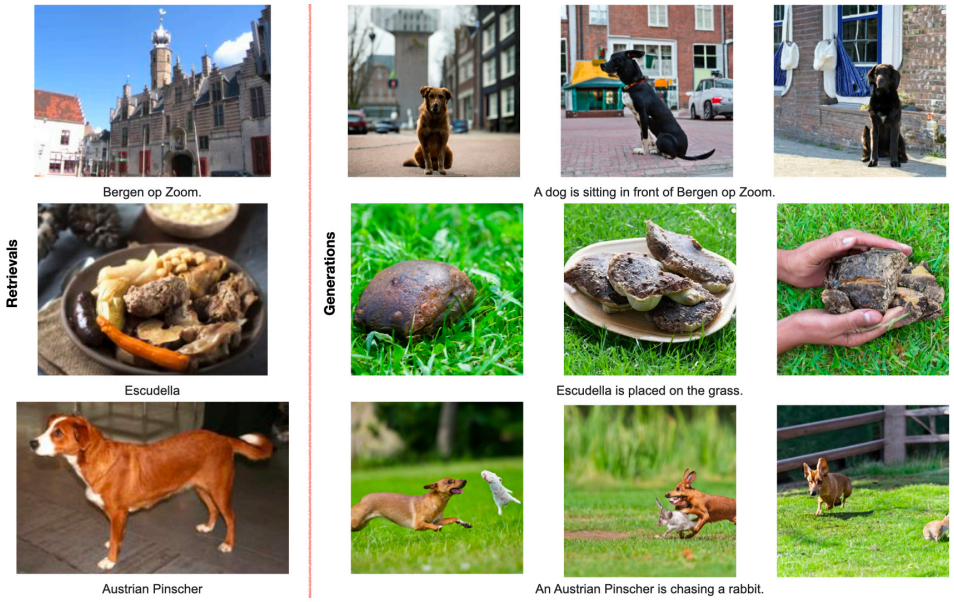

Figure 11: Failure examples from EntityDrawBench for dogs, landmarks, and foods.

- '[DOG] is fighting with another dog.',
- '[FOOD] is placed on the grass.',
- '[FOOD] is served with wine.',
- '[FOOD] with popcorn on the side.',
- 'A dog is beside [FOOD] (food).',
- '[FOOD] is decorated with flowers.'
- 'A dog is sitting in front of [LANDMARK].',
- 'A big crowd of tourists in front of [LANDMARK].',
- 'A rainy day in [LANDMARK].',
- '[LANDMARK] is lit up during the night.',
- 'cars parking in front of [LANDMARK].'

E FAILURE EXAMPLES

We found that Re-Imagen can also fail in a lot of cases. We demonstrate a few examples in Figure 11. As can be seen, the model sometimes has a few failure modes: (1) the text input prior is too strong like 'Zoom' will be interpreted as a 'Zoom-in' picture by the model. (2) the model cannot ground the retrieval text on the retrieval image, for example, the model believes that only the 'beef tenderloin inside the bowl' is 'Escudella' rather than the whole stew, therefore generating 'beef tenderloin on the grass'. (3) the model can sometimes mess up two conditions, for example, the reference 'Australian Pinscher' and the 'rabbit' in the prompt gets mixed into a single object.