Imagen Editor and EditBench: Advancing and Evaluating Text-Guided Image Inpainting

Su Wang* **Chitwan Saharia*** **Ceslee Montgomery***
Jordi Pont-Tuset **Shai Noy** **Stefano Pellegrini** **Yasumasa Onoe**
Sarah Laszlo **David J. Fleet** **Radu Soricut** **Jason Baldridge**
Mohammad Norouzi† **Peter Anderson†** **William Chan†**

Google Research

Figure 1. **A sequence of edits by Imagen Editor**. Given an *image*, a *user defined mask*, and a *text prompt*, Imagen Editor makes localized edits to the designated areas. The model meaningfully incorporates the user's intent and performs photorealistic edits.

Abstract

Text-guided image editing can have a transformative impact in supporting creative applications. A key challenge is to generate edits that are faithful to input text prompts, while consistent with input images. We present ***Imagen Editor****, a cascaded diffusion model built, by fine-tuning Imagen [36] on text-guided image inpainting. Imagen Editor's edits are faithful to the text prompts, which is accomplished by using object detectors to propose inpainting masks during training. In addition, Imagen Editor captures fine details in the input image by conditioning the cascaded pipeline on the original high resolution image. To improve qualitative and quantitative evaluation, we introduce* ***EditBench****, a systematic benchmark for text-guided image inpainting. EditBench evaluates inpainting edits on natural and generated images exploring objects, attributes, and scenes. Through extensive human evaluation on EditBench, we find that object-masking during training leads to across-the-board improvements in text-image alignment – such that Imagen Editor is preferred over DALL-E 2 [31] and Stable Diffusion [33] – and, as a cohort, these models are better at object-rendering than text-rendering, and handle material/color/size attributes better than count/shape attributes.*

1. Introduction

Text-to-image generation has seen a surge of recent interest [31, 33, 36, 50, 51]. While these generative models are surprisingly effective, users with specific artistic and design needs do not typically obtain the desired outcome in a single interaction with the model. Text-guided image *editing* can enhance the image generation experience by supporting interactive refinement [13, 17, 34, 46]. We focus on text-guided image inpainting, where a user provides an image, a masked area, and a text prompt and the model fills the masked area, consistent with both the prompt and the image context (Fig. 1). This complements mask-free editing [13, 17, 46] with the precision of localized edits [5, 27].

This paper contributes to the modeling and evaluation of text-guided image inpainting. Our modeling contribution is ***Imagen Editor***,[2] a text-guided image editor that combines large scale language representations with fine-grained control to produce high fidelity outputs. Imagen Editor is a cascaded diffusion model that extends Imagen [36] through finetuning for text-guided image inpainting. Imagen Editor adds image and mask context to each diffusion stage via three convolutional downsampling image encoders (Fig. 2).

A key challenge in text-guided image inpainting is ensuring that generated outputs are faithful to the text prompts. The standard training procedure uses randomly masked re-

*Equal contribution. †Equal advisory contribution.

[2] https://imagen.research.google/editor/

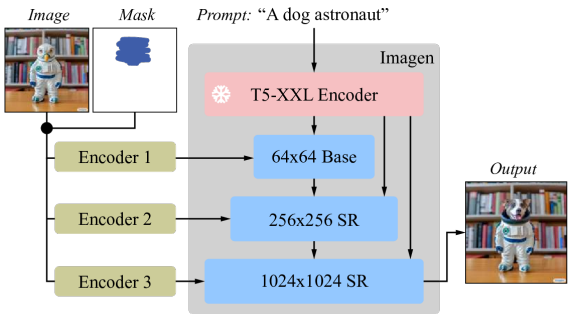

Figure 2. **Imagenator** is an image editing model built by fine-tuning Imagen. All of the diffusion models, i.e., the base model and super-resolution (SR) models, condition on high-resolution 1024×1024 image and mask inputs. To this end, new convolutional image encoders are introduced.

gions of input images [27, 35]. We hypothesize that this leads to weak image-text alignment since randomly chosen regions can often be plausibly inpainted using only the image context, without much attention to the prompt. We instead propose a novel *object masking* technique that encourages the model to rely more on the text prompt during training (Fig. 3). This helps make Imagen Editor more controllable and substantially improves text-image alignment.

Observing that there are no carefully-designed standard datasets for evaluating text-guided image inpainting, we propose ***EditBench***, a curated evaluation dataset that captures a wide variety of language, types of images, and levels of difficulty. Each EditBench example consists of (i) a masked input image, (ii) an input text prompt, and (iii) a high-quality output image that can be used as reference for automatic metrics. To provide insight into the relative strengths and weaknesses of different models, edit prompts are categorized along three axes: attributes (material, color, shape, size, count), objects (common, rare, text rendering), and scenes (indoor, outdoor, realistic, paintings).

Finally, we perform extensive human evaluations on EditBench, probing Imagen Editor alongside Stable Diffusion (SD) [33], and DALL-E 2 (DL2) [31]. Human annotators are asked to judge a) *text-image alignment* – how well the prompt is realized (both overall and assessing the presence of each object/attribute individually) and b) *image quality* – visual quality regardless of the text prompt. In terms of text-image alignment, Imagen Editor trained with object-masking is preferred in 68% of comparisons with its counterpart configuration trained with random masking (a commonly adopted method [27, 35, 41]). Improvements are across-the-board in all object and attribute categories. Imagen Editor is also preferred by human annotators relative to SD and DL2 (78% and 77% respectively). As a cohort, models are better at object-rendering than text-rendering, and handle material/color/size attributes better than count/shape attributes. Comparing automatic evaluation metrics with human judgments, we conclude that while

Figure 3. Random masks (left) frequently capture background or intersect object boundaries, defining regions that can be plausibly inpainted just from image context alone. *Object masks* (right) are harder to inpaint from image context alone, encouraging models to rely more on text inputs during training. (Note: This example image was generated by Imagen and is not in the training data.)

human evaluation remains indispensable, CLIPScore [14] is the most useful metric for hyperparameter tuning and model selection.

In summary, our main contributions are: (i) Imagen Editor, a new state-of-the-art diffusion model for high fidelity text-guided image editing (Sec. 3); (ii) EditBench, a manually curated evaluation benchmark for text-guided image inpainting that assesses fine-grained details such as object-attribute combinations (Sec. 4); and (iii) a comprehensive human evaluation on EditBench, highlighting the relative strengths and weaknesses of current models, and the usefulness of various automated evaluation metrics for text-guided image editing (Sec. 5).

2. Related Work

Text-Guided Image Editing. There has been much recent work on text-guided image inpainting [1, 3, 5, 7, 19, 27, 31, 33]. Paint By Word [3] optimizes for a balance between a) the consistency between the input and edited images, and b) the consistency between the text guide and the edited image. The technique has been used effectively more recently in DiffusionCLIP [18]. Blended Diffusion [1] runs CLIP-guided diffusion on the foreground (masked region) and the background (the context) in parallel and separately, and then blends the result by element-wise aggregation. CogView2 [7] proposes an auto-regressive text-guided infilling technique powered by cross-modal language modeling. DiffEdit [5] presents a "masked mask-free" formulation where masking segmentation and masked diffusion are run in parallel to apply masked inpainting. Most relevant to our work are Stable Diffusion [33] and GLIDE/DALL-E 2 [27, 31], which are also diffusion models. Key differences in our work are the use of an object detector for masking plus architectural changes to enable high resolution editing.

There has also been much work in ***mask-free*** text-guided image editing [2, 13, 17, 46]. Text2Live [2] operates on an isolated edit-layer with semantic localization, which allows

for good context preservation yet does not lend itself well to extensive modifications. Prompt-to-Prompt [13] presents powerful manipulation techniques on the cross-attention in the text-conditioning module. Imagic [17] optimizes a special embedding to capture the semantics of the input image, and produces textually faithful edits by interpolating the optimized embedding with the embedding of the target text.

Evaluation of Text-Guided Image Editing. Text-guided image inpainting has primarily been evaluated with respect to three aspects. (1) Image quality [1, 25, 27, 47] assesses the standalone quality of an image, usually independent of a (single) ground truth reference. (2) Reconstruction fidelity [10, 18, 23], on the other hand, calculates a similarity between the evaluated image and *a* ground truth. (3) Text-image alignment [28, 44, 53] measures similarity between visual outputs and textual inputs. (1) and (3) are most relevant to our work because text-guided image inpainting promotes diverse coverage (contingent on semantic coherence) rather than faithfulness to one particular reference.

Automatic evaluation. The standard automatic metric for image quality is *Frechét Inception Distance (FID)*, which assesses the quality of images in the latent space of a generative model with respect to the distribution of a set of real images. For text-image alignment, metrics based on text-image encoders (notably CLIP [30]) have been popular, e.g. CLIPScore [14] - distance between text and image encodings; CLIP-R-Precision [28] - the retrieval rank of the edited/synthesized image for the ground truth text among distractors. In this work, we further explore the connection between automatic and human evaluation, gauging the extent to which the automatic metrics agree with human assessments of model performance (for which there is currently no substitute when judging model outputs).

Human evaluation. The most typical formulation is asking two questions about side-by-side outputs from competing models – *which has the better image quality?*, and *which aligns with this* ${\texttt{text}}$ *better?* (paraphrased in varied ways). EditBench extends this paradigm by constructing a benchmark along diverse feature axes (attribute/object/scene), focusing the evaluation on the masked area rather than the full image (which delineates the evaluation of image *editing* from *generation*), and asking annotators to assess the presence of each object and attribute mentioned in the prompt *individually*. This distinguishes our work from the previous efforts in text-guided image inpainting, which typically evaluate in a less systematic manner or merely share cherry-picked examples [1, 18, 27, 53].

3. Imagen Editor

Imagen Editor is a text-guided image inpainting model targeting improved representation and reflection of linguistic inputs, fine-grained control and high fidelity outputs. Imagen Editor takes three inputs from the user, 1) the image to be edited, 2) a binary mask to specify the edit region, and 3) a text prompt – and all three inputs are used to guide the output samples. Imagen Editor is a diffusion-based model [6] fine-tuned from Imagen [36] for editing. See Figure 2 for an illustration.

Object Detector Masking Policy. A natural question to ask is: what kind of masks do we use to train models for text-guided image inpainting? The masked regions should be well aligned to the edit text prompt. Ideally, we would have a large expert dataset of aligned mask-prompt edits to train on; however, such a dataset does not exist and curating a large one would be difficult. One natural, simple policy to use is a random mask distribution, for example random box and/or random stroke masks; this has been successfully applied to prior inpainting models [35,48,49]. However, when random masks are used during training, they may cover a region irrelevant to the text prompt (Fig 3 left). Training on such examples can encourage the model to ignore the text prompt. We find this issue to be especially prevalent when masked regions are small or only partially cover an object, which was similarly observed for CogView2 [7].

Unlike simple text-**un**conditional inpainting, we need generated regions (from the mask) to not only be realistic, but also to relate coherently to the input text prompt. We propose a simple, effective solution to this problem. We hypothesize that masking out identified *objects* entirely will induce a greater overlap with the text prompt (Fig 3), and consequently encourage the model to pay more attention to the text prompt when inpainting. We use an off-the-shelf object detector to detect and localize objects, and use these bounding object boxes to generate masks to be used during training. The model we use is the lightweight SSD Mobilenet v2 [39] [3] which can be easily run on-the-fly and thus offers the same flexibility as random masking policies. Our experiments show that this simple modification to the masking policy works surprisingly well, and it alleviates most of the issues faced by models trained with a random masking policy. See the Appendix for implementation details.

High-Resolution Editing. In Imagen Editor, we modify Imagen to condition on both the image and the mask by concatenating them with the diffusion latents along the channel dimension, similar to SR3 [38], Palette [35] and GLIDE [27]. The conditioning image and the corresponding mask input to Imagen Editor are always at 1024×1024 resolution. The base diffusion 64×64 model and the $64\times64\rightarrow256\times256$ super-resolution model operate at a smaller resolution, and thus require some form of downsampling to match the diffusion latent resolution (e.g., 64×64 or 256×256). One method is to use a parameter-free downsampling operation (e.g., bicubic); we instead apply a parameterized downsampling convolution (e.g., convolution with a stride). In initial experiments we found this pa-

[3] https://tfhub.dev/tensorflow/ssd_mobilenet_v2/2

Mask Simple
"A metal cat"

Mask Rich
"A metal cat **is sitting and with its tail wrapped around its body**"

Full Image
"A metal cat **sitting in the middle of a farm field.**

Figure 4. **EditBench example**. The full image is used as a reference for successful inpainting. The mask covers the target object with a free-form, non-hinting shape. The three descriptions types are: single-attribute description of the masked object (**Mask-Simple**), multi-attribute description of the masked object (**Mask-Rich**), or whole image (**Full**). *Mask-Rich* especially probes models' ability to handle complex attribute binding and inclusion [12].

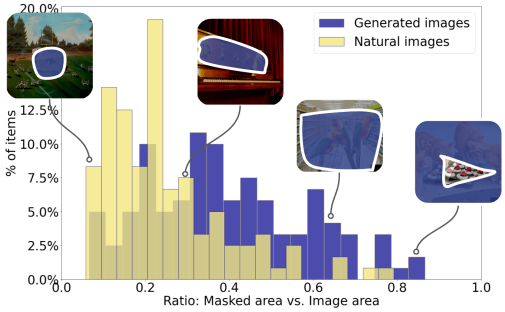

Figure 5. EditBench encompasses a wide variety of mask sizes, including large masks that contact the edges of the images (which can amount to an *uncropping* task in some cases).

rameterized downsampling operation to be critical for high fidelity. Simple bicubic downsampling resulted in significant artifacts along the mask boundaries in the final output image, and switching to a parameterized downsampling convolution resulted in much higher fidelity. We also initialize the corresponding new input channel weights to zero (like [27]) – at initialization the model is identical to Imagen (it ignores the conditioning image & mask).

Classifier-Free Guidance. Classifier-Free Guidance (CFG) [16] is a technique to bias samples to a particular conditioning (e.g., text prompt), at the cost of mode coverage. CFG has been found to be highly effective in boosting text-image alignment as well as image fidelity in text→image models [9, 27, 36, 50]. We found CFG continues to be critical for ensuring strong alignment between the generated image and the input text prompt for text-guided image inpainting. We follow [15] and use high guidance weights with guidance oscillation. In the base model, where ensuring strong alignment with text is most critical, we use a guidance weight schedule which oscillates between 1 and 30. We observe that high guidance weights combined with oscillating guidance [15] result in the best trade-off between sample fidelity and text-image alignment.

4. EditBench

Overview. EditBench is a new benchmark for text-guided image inpainting based on 240 images. Each image is paired with a mask that specifies the image region to be modified via inpainting. For each image-mask pair, we provide three different text prompts, representing different approaches to specifying the edit (see Fig. 4). Similar to the DrawBench [36] and PartiPrompts [50] benchmarks for text-to-image generation, EditBench is hand-curated to capture a wide variety of categories and aspects of difficulty.

Image Collection. EditBench includes both natural images drawn from existing computer vision datasets (Visual Genome [20] and Open Images [21]), and synthetic images generated by text-to-image models (Imagen [36] and Parti [50]) at 50:50 ratio. To construct EditBench, we first generate a wide variety of initial prompts to guide the image collection process. Initial prompts are generated by enumerating *attribute-object-scene* combos from these categories:

- *Attributes*: {*material, color, shape, size, count*};
- *Objects*: {*common, rare, text-rendering*};
- *Scenes*: {*indoor, outdoor, realistic, painting*}.

The choice of object, attribute and scene categories was inspired by studying image editing requests on Reddit.[4] Natural images are selected by manually searching for images matching object, attribute, scene combinations, e.g., for 'a=*material*|o=*common*|s=*outdoor*', images of an outdoor patio made of wood could be selected, instantiating 'a=*wooden*|o=*patio*|s=*outdoor*'. Synthetic images are created by sampling an object and attribute within each category (e.g., instantiating 'a=*material*|o=*common*|s=*outdoor*' as 'a=*metal*|o=*cat*|s=*outdoor*'), writing a matching prompt (e.g., *a metal cat standing in the middle of a farm field.*), sampling batches of images from text-to-image models as candidates, and then manually identifying the generated image that best matches the prompt. As shown in Figure 4, synthetic images can capture object-attribute-scene combinations that are unlikely to occur naturally, and editing these images is an important use case as part of the workflow of image creation combining descriptions and gesture.

Image Masks. For each image, we manually-annotate a free-form mask that completely covers the target object. We are careful not to too-closely segment the target object, which could leak information about the object underneath via its shape. We also include masks with a range of sizes (Fig. 5) to check the undesirable sensitivity to mask sizes [32]. We check models' robustness against the tendency of painting-over small masks due to overwhelming influence from the context; we also evaluate large-area inpainting and uncropping where the challenge is to not com-

[4] https://www.reddit.com/r/PhotoshopRequest/

pletely disregard the relatively small context.

Creating Text Prompts. Prior work (e.g., GLIDE [27]) often demonstrate text-guided image inpainting using prompts that describe the full image. However, writing full-image descriptions is unnatural for the use case of inpainting particular components of an image that depicts a complex scene with multiple objects and characters. Consequently, for each image-mask pair, we create three text prompts to probe model behavior from different angles. One type of prompt gives only a basic description (**Mask-Simple**) for the mask, another gives much more details (**Mask-Rich**), and the last describes the full image (**Full**, disregarding the mask). The unmasked input image itself serves as a reference image for inpainting in accordance with the prompts. See Fig. 4 for a summary.

5. Evaluation

We conduct comprehensive human evaluations of both text-image alignment and image quality on EditBench. We also analyze human preferences relative to automatic metrics. We evaluate four models:

- **Imagen Editor (IM)**: Our full model described in Sec. 3;
- **Imagen Editor$_{\text{RM}}$ (IM$_{\text{RM}}$)**: Imagen Editor finetuned with **R**andom **M**asking instead of object masking;
- **Stable Diffusion (SD)**: Version 1.5 of the model based on Rombach et al. [33];[5]
- **DALL-E 2 (DL2)**: A commercial web UI based on Ramesh et al. [31], accessed in October 2022.[6]

We compare Imagen Editor with Imagen Editor$_{\text{RM}}$ to quantify the benefits of object masking during training. We include evaluations of Stable Diffusion and DALL-E 2 to place our work in context with prior work and to more broadly analyze the limitations of the current SoTA.

5.1. Human Evaluation Protocol

We perform two types of human evaluations: *single image* evaluations and forced choice *side-by-side image* evaluations. The former allow us to ask fine-grained questions to establish if each individual object and attribute in the prompt has been correctly rendered. Side-by-side evaluations focus on comparisons between Imagen Editor and the other models, capturing *relative* model performance. We evaluate text-image alignment in both settings, and overall image quality only in side-by-side evaluations. In all evaluations we use a red box to highlight the image region edited by the model and ask the annotator to pay special attention to it (Fig. 6). Each model is evaluated based on four sampled image edits for each prompt.

Single Image Evaluations. Our single image evaluations are adapted to the given detail level for each type of prompt. For the ***Full*** prompts (describing the full image), annotators

[5] http://huggingface.co/runwayml/stable-diffusion-v1-5

[6] https://openai.com/dall-e-2/

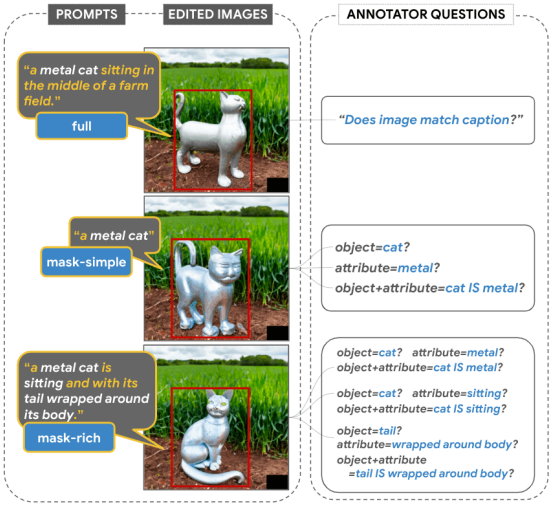

Figure 6. Human evaluation for single model text-image alignment. *Full* elicits annotators' overall impression of text-image alignment; *Mask-Simple* and *Mask-Rich* check for the correct inclusion of particular attributes and objects, and attribute binding.

assess general text-image alignment by giving a binary answer to the question *Does the image match the caption?*. For ***Mask-Simple*** prompts describing the masked region with one object and one attribute (e.g. *a metal cat*), evaluations are more fine-grained (Fig. 6). Annotators answer three binary questions, evaluating: (1) whether the object (*cat*) is rendered, (2) whether the given attribute (*metal*) is present in the image, and (3) whether the attribute (*metal*) is depicted applied to the correct object (*cat*) [12, 50]. Finally, for ***Mask-Rich*** prompts, we extend the previous evaluation to multiple attribute-object pairs. The annotator answers three sets of three binary questions – about the attribute, the object, and the attribute binding – making 9 binary judgments in total. Compared to previous evaluations in image generation that only assess general text-image alignment [1, 27, 53], our fine-grained evaluations provide greater insight into language fidelity and also which categories of objects and attributes present the most difficulties. Annotators in total perform 11.5K single model evaluation tasks (240 images $\times$ 3 prompts $\times$ 4 models $\times$ 4 samples).

Side-by-Side Evaluations. For ***Mask-Rich*** prompts, we extend the previous evaluation to multiple attribute-object pairs. The annotator answers three sets of three binary questions – about the attribute, the object, and the attribute binding – making 9 binary judgments in total. Compared to previous evaluations in image generation that only assess general text-image alignment [1, 27, 53], our fine-grained evaluations provide greater insight into language fidelity and also which categories of objects and attributes present the most difficulties. 18 (US-based) annotators in total perform 11.5K single model evaluation tasks (240 images $\times$ 3 prompts $\times$ 4 models $\times$ 4 samples). The Appendix has

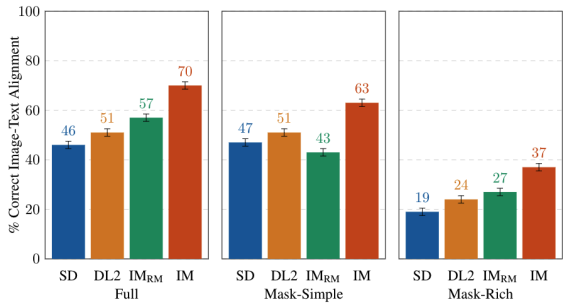

Figure 7. Single-image human evaluations of text-guided image inpainting on EditBench by *prompt type*. In this figure, for Mask-Simple and Mask-Rich prompts, text-image alignment is only counted as correct if the edited image correctly includes *every* attribute and object specified in the prompt, including the correct attribute binding (setting a very high bar for correctness). Note that due to different evaluation designs, Full vs Mask-only prompts results are less directly comparable.

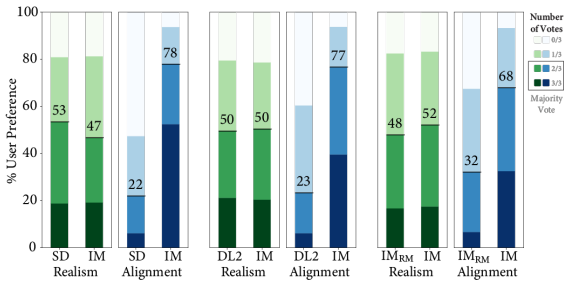

Figure 8. Side-by-side human evaluation of image realism & text-image alignment on EditBench Mask-Rich prompts. For text-image alignment, Imagen Editor is preferred in all comparisons.

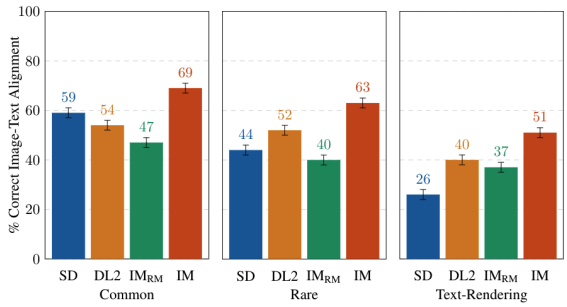

Figure 9. Single-image human evaluations on EditBench Mask-Simple by *object type*. As a cohort, models are better at object-rendering than text-rendering.

more details on the human evaluation process.

5.2. Human Evaluation Results

Overall. Fig. 7 presents the aggregated human ratings, sliced by prompt types. % Correct Image-Text Alignment is the proportion of positive judgments a model receives.

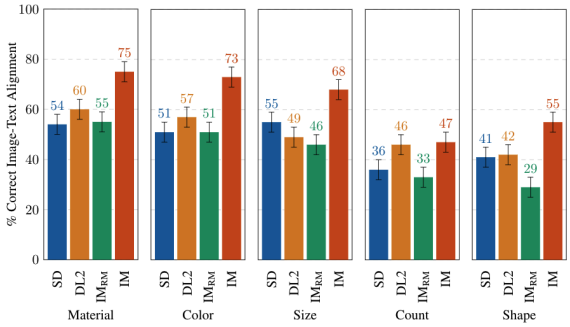

Figure 10. Single-image human evaluations on EditBench Mask-Simple by *attribute type*. Object masking improves adherence to prompt attributes across-the-board (IM vs. IM_{RM}).

Each question is binary – in the case of *Full*, responses reflect the overall impression, whereas for *Mask-Simple* and *Mask-Rich* a positive response indicates the edited image attributes are correctly bound to the correct objects. Across-the-board, Imagen Editor receives the highest ratings (10-13% higher than the 2nd highest). For the rest, the performance order is $\text{IM}_{\text{RM}} > \text{DL2} > \text{SD}$ (with 3-6% difference) except for with *Mask-Simple*, where IM_{RM} falls 4-8% behind. As relatively more semantic content is involved in *Full* and *Mask-Rich*, we conjecture IM_{RM} and IM are benefited by the higher performing T5 XXL text encoder (see [13], D1).

An interesting observation is annotators rate models higher with *Full* than *Mask-Simple* prompts, even though the former involves more semantic content. There are two likely reasons for this: a) models are trained by conditioning on *Full* prompts rather than mask-only [31, 33, 37]; b) since we do not change the context (unmasked), *Full* gains an advantage of having correct associations for this portion because the unmasked (and correct) pixels remain unchanged.

Finally, note that with *Mask-Rich*, while IM retains 10+% lead over the rest (see Fig. 7 and examples in Fig. 11), the overall performance drops substantially – leaving considerable room for future improvement.

Side-by-Side. In Fig. 8, compared with other models 1v1, IM leads in text alignment with a substantial margin, being preferred by annotators 78%, 77%, and 68% of the time compared to SD, DL2, and IM_{RM} respectively. These gains were realized while achieving similar levels of performance (0-6% delta) in image quality.

Breakdown by *objects*. In Fig. 9, IM leads in all object types: 10%, 11%, and 11% higher than the 2nd highest in *common*, *rare*, and *text-rendering*. For the rest, the notable observation is SD's performance plummets in text-rendering (59% & 44% for *common* and *rare*, and only 26% for *text-rendering*).

Breakdown by *attributes*. In Fig. 10, IM is rated much higher (13-16%) than the 2nd highest, except for in *count*, where DL2 is merely 1% behind. IM also improves the least

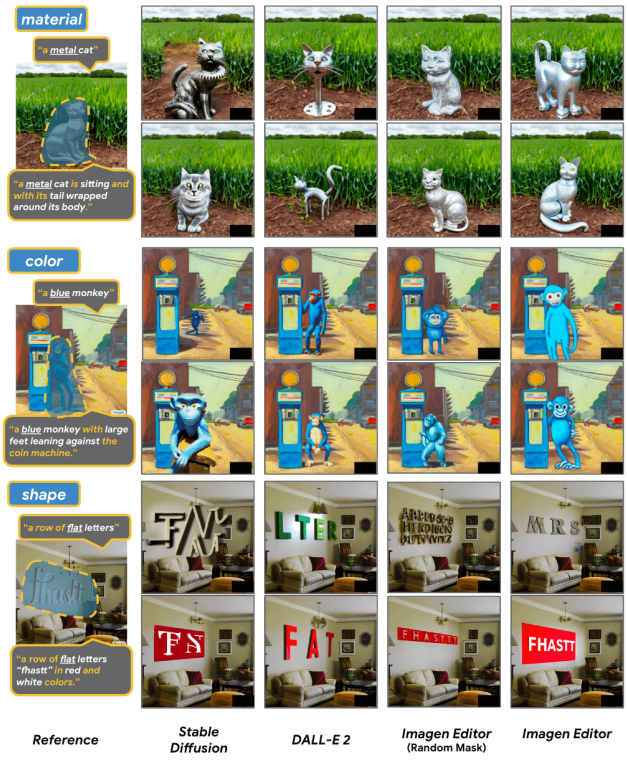

Figure 11. Example model outputs for *Mask-Simple* vs. *Mask-Rich* prompts. Object masking improves Imagen Editor's fine-grained adherence to the prompt compared to the same model trained with random masking. See Appendix for more examples.

over IM_{RM} in this attribute type (14%, with the 2nd lowest 22%), making *count* a particularly interesting category for future study. In general models get rated lower in *count* and *shape*, and the two happen to be the more abstract. The result is intuitive yet runs counter to findings in the recognition of abstract attributes [26]. The relatively similar performance in *size* vs. *material/count* is slightly unexpected – it is a relational category where the understanding requires appropriate contextualization, while for the latter the object itself is the only source of information wrt. other objects or the general surrounding. One possibility is the simplicity of our design: we include straightforward comparison of different sizes but do not build adversarial cases where seemingly small/large objects are in actuality the opposite.

5.3. Automatic Evaluation Metrics

Although human evaluations are widely adopted as the gold standard for image realism and text-image alignment evaluations, automatic evaluation metrics are valuable for iterative hyperparameter tuning and model selection. We compare human judgments with automatic metrics to identify the best metrics for model development.

Metrics. We investigate text-image alignment metrics based on **CLIPScore** [14] and **CLIP-R-Prec(ision)** [28].

Prompt	Image	T2I	I2I	T2I+I2I	R-Prec	Rand
Full	Full	**70.1**	58.6	66.8	53.3	50.0
Full	Crop	68.1	55.8	62.4	57.7	50.0
Mask-Simple	Full	73.8	53.1	63.2	72.0	50.0
Mask-Simple	Crop	**76.0**	55.3	66.4	71.0	50.0
Mask-Rich	Full	66.7	55.2	63.4	62.3	50.0
Mask-Rich	Crop	**68.4**	56.4	64.1	63.3	50.0

Table 1. Percentage agreement between CLIPScore metrics and human judgments when picking the *best image out of two model-generated images* for the same text prompt. Text-to-image (T2I) CLIPScore similarity outperforms CLIP-R-precision (R-Prec) and image-to-image (I2I) similarity using a reference image.

Prompt	Image	T2I	I2I	T2I+I2I	R-Prec	Rand
Full	Full	**38.5**	30.8	35.7	28.7	25.0
Full	Crop	36.7	28.2	32.4	27.9	25.0
Mask-Simple	Full	45.7	28.2	37.1	40.2	25.0
Mask-Simple	Crop	**47.1**	29.4	38.5	39.7	25.0
Mask-Rich	Full	45.8	31.4	40.5	39.1	25.0
Mask-Rich	Crop	**48.1**	30.9	40.9	39.3	25.0

Table 2. Agreement between CLIPScore metrics and human judgments when repeatedly picking the *best model out of four hybrid models*. Text-to-image (T2I) similarity outperforms CLIP-R-precision (R-Prec) and reference image-to-image (I2I) similarity.

CLIPScore calculates text-to-image (T2I) or image-to-image (I2I) similarity in the latent space of the contrastively-trained CLIP model [29]. CLIP-R-Prec is a ranking based approach (typically formulated as text R-Precision [28]) that measures how well the generated image retrieves the text prompt with CLIP from among a set of text distractors. As text distractors we use all the other prompts in EditBench with the same prompt type.

Comparison to human judgments. To evaluate agreement between automatic metrics and human scores, a common practice is to report correlation coefficients [40]. However, metrics such as Spearman's ρ consider the ranking induced by each score over *all pairs* of observations, which includes ranking images with *different prompts*. Rather than comparing images with different prompts, which is a difficult judgment even for people, we focus on two questions: (1) For a given prompt, can automatic metrics pick the image preferred by people? and (2) Can automatic metrics identify the model with the highest human evaluations?

In Tab. 1 we report the agreement between various metrics based on CLIPScore and human judgments when picking the *best image* from two model-generated images with the same text prompt. We sample 10K image pairs and the best image in each pair is determined by human single-image evaluation scores (image pairs with the same human score are excluded). Metrics are calculated using both

	SD	DL2	IM_{RM}	IM	Ref.
CLIPScore (↑)					
T2I	29.7	29.1	29.6	**31.5**	31.0
I2I	74.9	76.1	75.8	**76.6**	-
T2I+I2I	52.3	52.6	53.1	**53.6**	-
CLIP-R-Prec (↑)	96.5	95.3	95.0	**98.6**	99.3
NIMA (↑)	4.44	4.33	4.56	**4.63**	4.89

Table 3. Aggregated automated metric scores. **boldface**: highest scoring model; box: reference images rated highest.

the full image (Full) and a cropped bounding box around the masked region (Crop). We find that CLIPScore based on text-to-image (T2I) similarity has the highest agreement with human judgments, identifying the best image in 68-76% of pairs, depending on the prompt type. Unsurprisingly, CLIPScore has higher agreement with humans on simpler prompts (Mask-Simple) compared to more complex prompts (Mask-Rich).

In Tab. 2 we report agreement with human judgments when picking the *best model* based on evaluations aggregated across EditBench. Each evaluation is a choice between 4 hybrid models created by randomly selecting one sampled image from one of the available models for each prompt [11, 52]. Scores for each hybrid model are created by averaging the scores for the corresponding images in the sampled data, and 100K evaluations are performed. Similarly to Tab. 1, we find that CLIPScore (T2I) is most reliable metric (identifying the best hybrid model out of 4 in 39-48% of instances). In both experiments CLIPScore works best when the image region matches the prompt, i.e., full image (Full) when the prompt describes the full image, and cropped bounding box around the masked region (Crop) when the text prompt describes only the masked region. In both Tab. 1 and Tab. 2 the 95% confidence intervals calculated with bootstrap resampling are below 1%.

Overall results. In Tab. 3 we report automatic metrics aggregated over all prompts for each model, and for the EditBench reference images. For CLIPScore metrics, the image representation (Full or Crop) is aligned with the prompt (Full or Mask). We also report **NIMA** [42] – a model-based perceptual image quality metric. We find that the reference images receive the highest CLIP-R-Precision. Imagen Editor is ranked highest among the 4 models on each metric.

6. Societal Impact

The image editing models presented are part of the growing family of generative models which unlock new capabilities in content creation, however, they also have the potential to create content that is harmful to individuals or to society. In language modeling, it is now well recognized [43, 45] that text generation models are prone to recapitulating and amplifying social biases that may be present in their training sets. The risk of amplifying social harm also pertains to text-to-image generation and text-guided image inpainting; as discussed elsewhere, the data used to train these models is equally fraught [4, 36, 50].

A particular risk that is exposed by text-guided image inpainting, but is not present in text-to-image models is that inpainting might enable the scaled and simple creation of convincing misinformation– for example, editing an image of a political figure to include a controlled substance. Two approaches to mitigating this risk that we have taken in our experimentation thus far are to (1) ensuring that distinctive watermarks are present on each generated image, and (2) refraining from photorealistic generation of human faces. To extend the protections against misinformation, robust methods for proving image provenance such as steganographic watermarking are helpful [24]. In addition, de-duplication of text-image training datasets can reduce the likelihood that a model reproduces a training set image [22].[7] Robust guardrails are needed to prevent recognizable likenesses of people from being generated and exposed to users.

7. Conclusion

We presented Imagen Editor and EditBench, making significant advancements in text-guided image inpainting and the evaluation thereof. Imagen Editor is a text-guided image inpainting finetuned from Imagen. Key to Imagen Editor is adding new convolution layers to enable high-resolution editing, and the use of a object masking policy for training. EditBench is a comprehensive systematic benchmark for text-guided image inpainting. EditBench systematically evaluates text-guided image inpainting across multiple dimensions: attributes, objects, and scenes. We find Imagen Editor to outperform DALL-E 2 and Stable Diffusion on EditBench in both human evaluation and automatic metrics.

Acknowledgment. We would like to thank Gunjan Baid, Nicole Brichtova, Sara Mahdavi, Kathy Meier-Hellstern, Zarana Parekh, Anusha Ramesh, Tris Warkentin, Austin Waters, Vijay Vasudevan for their generous help through the course of the project. We give thanks to Igor Karpov, Isabel Kraus-Liang, Raghava Ram Pamidigantam, Mahesh Maddinala, and all the anonymous human annotators for assisting us to coordinate and complete the human evaluation tasks. We are grateful to Huiwen Chang, Austin Tarango, Douglas Eck for reviewing the paper and providing feedback. Thanks to Erica Moreira and Victor Gomes for help with resource coordination. Finally, we would like to give our thanks and appreciation to the authors of DALL-E 2 [31] for their permission for us to use the outputs from their model for research purposes.

[7] https://openai.com/blog/dall-e-2-pre-training-mitigations/

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. *Proceedings of CVPR*, abs/2111.14818, 2022. 2, 3, 5, 12

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing, 2022. 2

[3] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *CoRR*, abs/2103.10951, 2021. 2

[4] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. In *arXiv:2110.01963*, 2021. 8

[5] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance, 2022. 1, 2

[6] Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[7] Ming Ding, Wendi Zheng, Wenyi Hong, and Jie Tang. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *arXiv preprint arXiv:2204.14217*, 2022. 2, 3

[8] Sara Dolnicar, Bettina Grün, and Friedrich Leisch. Quick, simple and reliable: forced binary survey questions. *International Journal of Market Research*, 2011. 12

[9] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022. 4

[10] Ying Gao and Qing Zhu. Text-guided image inpainting. In *Proceedings of IEEE*, 2022. 3

[11] Yvette Graham and Qun Liu. Achieving accurate conclusions in evaluation of automatic machine translation metrics. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016. 8

[12] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *CoRR*, abs/2012.05208, 2020. 4, 5

[13] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-Prompt Image Editing with Cross Attention Control. In *arXiv preprint arXiv:2208.01626*, 2022. 1, 2, 3, 6

[14] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021. 2, 3, 7

[15] 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. 4

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 4

[17] 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, 2022. 1, 2, 3

[18] Gwanghyun Kim and Jong Chul Ye. Diffusionclip: Text-guided image manipulation using diffusion models. *arXiv preprint arXiv:2110.02711*, 2021. 2, 3, 12

[19] Yoon Kim and Alexander M. Rush. Sequence-Level Knowledge Distillation. In *EMNLP*, 2016. 2

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 2017. 4

[21] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 4

[22] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. *CoRR*, abs/2107.06499, 2021. 8

[23] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H. S. Torr. Manigan: Text-guided image manipulation, 2020. 3

[24] Xiyang Luo, Michael Goebel, Elnaz Barshan, and Feng Yang. Leca: A learned approach for efficient cover-agnostic watermarking, 2022. 8

[25] Naila Murray, Luca Marchesotti, and Florent Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *Proceedings of CVPR*, 2012. 3

[26] Zhixiong Nan, Yang Liu, Nanning Zheng, and Song-Chun Zhu. Recognizing unseen attribute-object pair with generative model. *Proceedings of AAAI*, 2019. 7

[27] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Bob McGrew Pamela Mishkin, Ilya Sutskever, and Mark Chen. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In *arXiv:2112.10741*, 2021. 1, 2, 3, 4, 5, 12

[28] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 3, 7

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 7

[30] 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 vi-

sual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. 3

[31] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. In *arXiv:2204.06125*, 2022. 1, 2, 5, 6, 8

[32] Walber Rodrigues, Felipe Walmsley, George Cavalcanti, Jonysberg Quintino, and Helder Pinho. Grave artifacts in image inpainting: Investigating the causes and untangling the factors, 2021. 4

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*, 2022. 1, 2, 5, 6

[34] 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. 2022. 1

[35] Chitwan Saharia, William Chan, Huiwen Chang, Chris A. Lee, Jonathan Ho, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Palette: Image-to-Image Diffusion Models. In *arXiv:2111.05826*, 2021. 2, 3, 13

[36] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In *NeurIPS*, 2022. 1, 3, 4, 8

[37] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 6

[38] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image Super-Resolution via Iterative Refinement. *IEEE PAMI*, 2022. 3

[39] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018. 3

[40] Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André F. T. Martins. Findings of the WMT 2021 shared task on quality estimation. In *Proceedings of the Sixth Conference on Machine Translation*, 2021. 7

[41] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. *arXiv preprint arXiv:2109.07161*, 2021. 2

[42] Hossein Talebi, Ehsan Amid, Peyman Milanfar, and Manfred K. Warmuth. Rank-smoothed pairwise learning in perceptual quality assessment. In *Proceedings of IEEE*, 2021. 8

[43] Yi Chern Tan and L. Elisa Celis. Assessing Social and Intersectional Biases in Contextualized Word Representations. In *NeurIPS*, 2019. 8

[44] Ming Tao, Bing-Kun Bao, Hao Tang, Fei Wu, Longhui Wei, and Qi Tian. De-net: Dynamic text-guided image editing adversarial networks, 2022. 3

[45] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Kathleen S. Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed H. Chi, and Quoc Le. Lamda: Language models for dialog applications. *CoRR*, abs/2201.08239, 2022. 8

[46] Dani Valevski, Matan Kalman, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning an image generation model on a single image, 2022. 1, 2

[47] Jianan Wang, Guansong Lu, Hang Xu, Zhenguo Li, Chunjing Xu, and Yanwei Fu. Manitrans: Entity-level text-guided image manipulation via token-wise semantic alignment and generation, 2022. 3

[48] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018. 3

[49] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4471–4480, 2019. 3, 13

[50] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling Autoregressive Models for Content-Rich Text-to-Image Generation. In *arXiv:2206.10789*, 2022. 1, 4, 5, 8

[51] Han Zhang, Weichong Yin, Yewei Fang, Lanxin Li, Boqiang Duan, Zhihua Wu, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vilg: Unified generative pre-training for bidirectional vision-language generation. *CoRR*, abs/2112.15283, 2021. 1

[52] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *ICLR*, 2020. 8

[53] Yufan Zhou, Ruiyi Zhang, Jiuxiang Gu, Chris Tensmeyer, Tong Yu, Changyou Chen, Jinhui Xu, and Tong Sun. Interactive image generation with natural-language feedback. In *Proceedings of AAAI*, 2022. 3, 5, 12