InstructPix2Pix: Learning to Follow Image Editing Instructions

Tim Brooks* Aleksander Holynski* Alexei A. Efros
University of California, Berkeley

Figure 1. Given an image and an instruction for how to edit that image, our model performs the appropriate edit. Our model does not require full descriptions for the input or output image, and edits images in the forward pass without per-example inversion or fine-tuning.

Abstract

We propose a method for editing images from human instructions: given an input image and a written instruction that tells the model what to do, our model follows these instructions to edit the image. To obtain training data for this problem, we combine the knowledge of two large pretrained models—a language model (GPT-3) and a text-to-image model (Stable Diffusion)—to generate a large dataset of image editing examples. Our conditional diffusion model, InstructPix2Pix, is trained on our generated data, and generalizes to real images and user-written instructions at inference time. Since it performs edits in the forward pass and does not require per-example fine-tuning or inversion, our model edits images quickly, in a matter of seconds. We show compelling editing results for a diverse collection of input images and written instructions.

1. Introduction

We present a method for teaching a generative model to follow human-written instructions for image editing. Since training data for this task is difficult to acquire at scale, we propose an approach for generating a paired dataset that combines multiple large models pretrained on different modalities: a large language model (GPT-3 [7]) and a text-to-image model (Stable Diffusion [51]). These two models capture complementary knowledge about language and images that can be combined to create paired training data for a task spanning both modalities.

Using our generated paired data, we train a conditional diffusion model that, given an input image and a text instruction for how to edit it, generates the edited image. Our model directly performs the image edit in the forward pass, and does not require any additional example images, full descriptions of the input/output images, or per-example fine-tuning. Despite being trained entirely on synthetic examples (i.e., both generated written instructions and generated
imagery), our model achieves zero-shot generalization to both arbitrary real images and natural human-written instructions. Our model enables intuitive image editing that can follow human instructions to perform a diverse collection of edits: replacing objects, changing the style of an image, changing the setting, the artistic medium, among others. Selected examples can be found in Figure 1.

2. Prior work

Composing large pretrained models Recent work has shown that large pretrained models can be combined to solve multimodal tasks that no one model can perform alone, such as image captioning and visual question answering (tasks that require the knowledge of both a large language model and a text-image model). Techniques for combining pretrained models include joint finetuning on a new task [4, 33, 40, 67], communication through prompting [62, 69], composing probability distributions of energy-based models [11, 37], guiding one model with feedback from another [61], and iterative optimization [34]. Our method is similar to prior work in that it leverages the complementary abilities of two pretrained models—GPT-3 [7] and Stable Diffusion [51]—but differs in that we use these models to generate paired multi-modal training data.

Diffusion-based generative models Recent advances in diffusion models [59] have enabled state-of-the-art image synthesis [10, 18, 19, 53, 55, 60] as well as generative models of other modalities such as video [21, 58], audio [30], text [35] and network parameters [45]. Recent text-to-image diffusion models [41, 48, 51, 54] have shown to generate realistic images from arbitrary text captions.

Generative models for image editing Image editing models traditionally targeted a single editing task such as style transfer [15, 16] or translation between image domains [22, 24, 36, 42, 71]. Numerous editing approaches invert [1–3, 12] or encode [8, 50, 63] images into a latent space (e.g., StyleGAN [25, 26]) where they can be edited by manipulating latent vectors. Recent models have leveraged CLIP [47] embeddings to guide image editing using text [5, 9, 14, 28, 31, 41, 44, 70]. We compare with one of these methods, Text2Live [6], an editing method that optimizes for an additive image layer that maximizes a CLIP similarity objective.

Recent works have used pretrained text-to-image diffusion models for image editing [5, 17, 27, 38, 48]. While some text-to-image models natively have the ability to edit images (e.g., DALL-E-2 can create variations of images, inpaint regions, and manipulate the CLIP embedding [48]), using these models for targeted editing is non-trivial, because in most cases they offer no guarantees that similar text prompts will yield similar images. Recent work by Hertz et al. [17] tackles this issue with Prompt-to-Prompt, a method for assimilating the generated images for similar text prompts, such that isolated edits can be made to a generated image. We use this method in generating training data. To edit non-generated (i.e., real) imagery, SDEdit [38] uses a pretrained model to noise and denoise an input image with a new target prompt. We compare with SDEdit as a baseline. Other recent works perform local inpainting given a caption and user-drawn mask [5, 48], generate new images of a specific object or concept learned from a small collection of images [13, 52], or perform editing by inverting (and fine-tuning) a single image, and subsequently regenerating with a new text description [27]. In contrast to these approaches, our model takes only a single image and an instruction for how to edit that image (i.e., not a full description of any image), and performs the edit directly in the forward pass without need for a user-drawn mask, additional images, or per-example inversion or finetuning.

Learning to follow instructions Our method differs from existing text-based image editing works [6, 13, 17, 27, 38, 52] in that it enables editing from instructions that tell the model what action to perform, as opposed to text labels, captions or descriptions of input/output images. A key benefit of following editing instructions is that the user can just tell the model exactly what to do in natural written text. There is no need for the user to provide extra information, such as example images or descriptions of visual content that remains constant between the input and output images. Instructions are expressive, precise, and intuitive to write, allowing the user to easily isolate specific objects or visual attributes to change. Our goal to follow written image editing instructions is inspired by recent work teaching large language models to better follow human instructions for language tasks [39, 43, 68].

Training data generation with generative models Deep models typically require large amounts of training data. Internet data collections are often suitable, but may not exist in the form necessary for supervision, e.g., paired data of particular modalities. As generative models continue to improve, there is growing interest in their use as a source of cheap and plentiful training data for downstream tasks [32, 46, 49, 57, 64, 65]. In this paper, we use two different off-the-shelf generative models (language, text-to-image) to produce training data for our editing model.

3. Method

We treat instruction-based image editing as a supervised learning problem: (1) first, we generate a paired training dataset of text editing instructions and images before/after the edit (Sec. 3.1, Fig. 2a-c), then (2) we train an image editing diffusion model on this generated dataset (Sec. 3.2, Fig 2d). Despite being trained with generated images and
3.1. Generating a Multi-modal Training Dataset

We combine the abilities of two large-scale pretrained models that operate on different modalities—a large language model [7] and a text-to-image model [51]—to generate a multi-modal training dataset containing text editing instructions and the corresponding images before and after the edit. In the following two sections, we describe in detail the two steps of this process. In Section 3.1.1, we describe the process of fine-tuning GPT-3 [7] to generate a collection of text edits: given a prompt describing an image, produce a text instruction describing a change to be made and a prompt describing the image after that change (Figure 2a). Then, in Section 3.1.2, we describe the process of converting the two text prompts (i.e., before and after the edit) into a pair of corresponding images using a text-to-image model [51] (Figure 2b).

3.1.1 Generating Instructions and Paired Captions

We first operate entirely in the text domain, where we leverage a large language model to take in image captions and produce editing instructions and the resulting text captions after the edit. For example, as shown in Figure 2a, provided the input caption “photograph of a girl riding a horse”, our language model can generate both a plausible edit instruction “have her ride a dragon” and an appropriately modified output caption “photograph of a girl riding a dragon”. Operating in the text domain enables us to generate a large and diverse collection of edits, while maintaining correspondence between the image changes and text instructions.

Our model is trained by finetuning GPT-3 on a relatively small human-written dataset of editing triplets: (1) input captions, (2) edit instructions, (3) output captions. To produce the fine-tuning dataset, we sampled 700 input captions from the LAION-Aesthetics V2 6.5+ [56] dataset and manually wrote instructions and output captions. See Table 1a for examples of our written instructions and output captions. Using this data, we fine-tuned the GPT-3 Davinci model for a single epoch using the default training parameters. Benefiting from GPT-3’s immense knowledge and ability to generalize, our finetuned model is able to generate creative yet sensible instructions and captions. See Table 1b for example GPT-3 generated data. Our dataset is created by generating a large number of edits and output captions using this trained model, where the input captions are real image captions from LAION-Aesthetics (excluding samples with duplicate captions or duplicate image URLs). We chose the LAION dataset due to its large size, diversity of content (including references to proper nouns and popular culture), and variety of mediums (photographs, paintings, digital artwork). A potential drawback of LAION is that it is quite noisy and contains a number of nonsensical or nondescriptive captions—however, we found that dataset noise is mitigated through a combination of dataset filtering (Section 3.1.2) and classifier-free guidance (Section 3.2.1). Our final corpus of generated instructions and captions consists of 454,445 examples.

3.1.2 Generating Paired Images from Paired Captions

Next, we use a pretrained text-to-image model to transform a pair of captions (referring to the image before and after the edit) into a pair of images. One challenge in turning a pair of captions into a pair of corresponding images is that text-to-image models provide no guarantees about image consistency, even under very minor changes of the conditioning
Table 1. We label a small text dataset, finetune GPT-3, and use that finetuned model to generate a large dataset of text triplets. As the input caption for both the labeled and generated examples, we use real image captions from LAION. Highlighted text is generated by GPT-3.

<table>
<thead>
<tr>
<th>Human-written (700 edits)</th>
<th>GPT-3 generated (&gt;450,000 edits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input LAION caption</td>
<td>Edit instruction</td>
</tr>
<tr>
<td>Yefim Volkov; Misty Morning</td>
<td>make it afternoon</td>
</tr>
<tr>
<td>girl with horse at sunset</td>
<td>change the background to a city</td>
</tr>
<tr>
<td>painting-of-forest-and-pond</td>
<td>Without the water.</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Alex Hill, Original oil painting on canvas, Moonlight Bay</td>
<td>in the style of a coloring book</td>
</tr>
<tr>
<td>The great elf city of Rivendell, sitting atop a waterfall as cascades of water spill around it</td>
<td>Add a giant red dragon</td>
</tr>
<tr>
<td>Kate Hudson arriving at the Golden Globes 2015</td>
<td>make her look like a zombie</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Figure 3. Pair of images generated using StableDiffusion [51] with and without Prompt-to-Prompt [17]. For both, the corresponding captions are "photograph of a girl riding a horse" and "photograph of a girl riding a dragon".

While this greatly helps assimilate generated images, different edits may require different amounts of change in image-space. For instance, changes of larger magnitude, such as those which change large-scale image structure (e.g., moving objects around, replacing with objects of different shapes), may require less similarity in the generated image pair. Fortunately, Prompt-to-Prompt has as a parameter that can control the similarity between the two images: the fraction of denoising steps \( p \) with shared attention weights. Unfortunately, identifying an optimal value of \( p \) from only the captions and edit text is difficult. We therefore generate 100 sample pairs of images per caption-pair, each with a random \( p \sim U(0.1, 0.9) \), and filter these samples by using a CLIP-based metric: the directional similarity in CLIP space as introduced by Gal et al. [14]. This metric measures the consistency of the change between the two images (in CLIP space) with the change between the two image captions. Performing this filtering not only helps maximize the diversity and quality of our image pairs, but also makes our data generation more robust to failures of Prompt-to-Prompt and Stable Diffusion.

3.2. InstructPix2Pix

We use our generated training data to train a conditional diffusion model that edits images from written instructions. We base our model on Stable Diffusion, a large-scale text-to-image latent diffusion model.

Diffusion models [59] learn to generate data samples through a sequence of denoising autoencoders that estimate the score [23] of a data distribution (a direction pointing toward higher density data). Latent diffusion [51] improves the efficiency and quality of diffusion models by operating in the latent space of a pretrained variational autoencoder [29] with encoder \( E \) and decoder \( D \). For an image \( x \), the diffusion process adds noise to the encoded latent \( z = E(x) \) producing a noisy latent \( z_t \) where the noise level increases over timesteps \( t \in T \). We learn a network \( e_\theta \) that predicts the noise added to the noisy latent \( z_t \) given image conditioning \( c_I \) and text instruction conditioning \( c_T \). We minimize the following latent diffusion objective:

\[
L = \mathbb{E}_{E(x), E(c_I), c_T, c \sim N(0, 1), t}(\|e - e_\theta(z_t, t, E(c_I), c_T)\|^2_2)
\]

Wang et al. [66] show that fine-tuning a large image diffusion models outperforms training a model from scratch for image translation tasks, especially when paired training data is limited. We therefore initialize the weights of our model with a pretrained Stable Diffusion checkpoint, lever-
agging its vast text-to-image generation capabilities. To support image conditioning, we add additional input channels to the first convolutional layer, concatenating $z_t$ and $E(c_I)$. All available weights of the diffusion model are initialized from the pretrained checkpoints, and weights that operate on the newly added input channels are initialized to zero. We reuse the same text conditioning mechanism that was originally intended for captions to instead take as input the text edit instruction $c_T$. Additional training details are provided in Appendix C of the supplement.

### 3.2.1 Classifier-free Guidance for Two Conditionings

Classifier-free diffusion guidance [20] is a method for trading off the quality and diversity of samples generated by a diffusion model. It is commonly used in class-conditional and text-conditional image generation to improve the visual quality of generated images and to make sampled images better correspond with their conditioning. Classifier-free guidance effectively shifts probability mass toward data where an implicit classifier $p_y(c|z_t)$ assigns high likelihood to the conditioning $c$. The implementation of classifier-free guidance involves jointly training the diffusion model for conditional and unconditional denoising, and combining the two score estimates at inference time. Training for unconditional denoising is done by simply setting the conditioning to a fixed null value $c = \emptyset$ at some frequency during training. At inference time, with a guidance scale $s \geq 1$, the modified score estimate $\tilde{e}_\theta(z_t, c)$ is extrapolated in the direction toward the conditional $e_\theta(z_t, c)$ and away from the unconditional $e_\theta(z_t, \emptyset)$.

$$
\tilde{e}_\theta(z_t, c) = e_\theta(z_t, \emptyset) + s \cdot (e_\theta(z_t, c) - e_\theta(z_t, \emptyset))
$$

(2)

For our task, the score network $e_\theta(z_t, c_I, c_T)$ has two conditionings: the input image $c_I$ and text instruction $c_T$. We find it beneficial to leverage classifier-free guidance with respect to both conditions. Liu et al. [37] demonstrate that a conditional diffusion model can compose score estimates from multiple different conditioning values. We apply the same concept to our model with two separate conditioning inputs. During training, we randomly set only $c_I = \emptyset_I$ for 5% of examples, only $c_T = \emptyset_T$ for 5% of examples, and both $c_I = \emptyset_I$ and $c_T = \emptyset_T$ for 5% of examples. Our model is therefore capable of conditional or unconditional denoising with respect to both or either conditional inputs. We introduce two guidance scales, $s_I$ and $s_T$. Increasing $s_I$ results in edited images that more closely resemble the input image, and increasing $s_T$ results in more intense edits. Our modified score estimate is as follows:

$$
e_\theta(z_t, c_I, c_T) = e_\theta(z_t, \emptyset, \emptyset) + s_I \cdot (e_\theta(z_t, c_I, \emptyset) - e_\theta(z_t, \emptyset, \emptyset)) + s_T \cdot (e_\theta(z_t, c_I, c_T) - e_\theta(z_t, c_I, \emptyset))
$$

(3)

In Figure 4, we show the effects of these two parameters on generated samples. See Appendix D in the supplement for details of our classifier-free guidance formulation.

### 4. Results

We show instruction-based image editing results on a diverse set of real photographs and artwork, for many edit types and instruction wordings. See Figures 1, 5, 6, 7, 11, 12 and Appendix A in the supplement for selected results. Our model successfully performs many challenging edits, including replacing objects, changing seasons and weather, replacing backgrounds, modifying material attributes, converting artistic medium, and a variety of others.

We compare our method qualitatively with recent works SDEdit [38], Text2Live [6], and Prompt-to-Prompt [17]. Our model follows instructions for how to edit the image, but prior works (including these baseline methods) expect descriptions of the image (or edit layer). Therefore, we provide them with the “after-edit” text caption instead of the edit instruction. We also compare our method quantitatively with SDEdit and Prompt-to-Prompt, using two metrics measuring image consistency and edit quality, further described in Section 4.1. Finally, we show ablations on how the size and quality of generated training data affect our model’s performance in Section 4.2.
Figure 5. *Mona Lisa* transformed into various artistic mediums.

Figure 6. *The Creation of Adam* with new context and subjects (generated at 768 resolution).

Figure 7. The iconic Beatles *Abbey Road* album cover transformed in a variety of ways.
4.1. Baseline comparisons

We provide qualitative comparisons with SDEdit [38], Text2Live [6], and Prompt-to-Prompt [17], as well as quantitative comparisons with SDEdit and Prompt-to-Prompt. SDEdit [38] is a technique for editing images with a pretrained diffusion model, where a partially noised image is passed as input and denoised to produce a new edited image. Text2Live [6] edits images by generating a color+opacity augmentation layer, conditioned on a text prompt.

We compare with SDEdit, Text2Live, and Prompt-to-Prompt qualitatively in Figure 9. Additional comparisons on other examples, as well as other configurations of these related works are provided in Appendix B of the supplement. We notice that while SDEdit works reasonably well for cases where content remains approximately constant and style is changed, it struggles to preserve identity and isolate individual objects, especially when larger changes are desired. Additionally, it requires a full output description of the desired image, rather than an editing instruction. On the other hand, while Text2Live is able to produce convincing results for edits involving additive layers, its formulation limits the categories of edits that it can handle.

Quantitative comparisons with SDEdit and Prompt-to-Prompt are shown in Figure 8. We plot the tradeoff between two metrics, cosine similarity of CLIP image embeddings (how much the edited image agrees with the input image) and the directional CLIP similarity introduced by [14] (how much the change in text captions agrees with the change in the images). These are competing metrics—increasing the degree to which the output correspond to a desired edit will reduce its similarity with the input image—and we are interested in which method achieves the best tradeoff (highest curve). We find that compared to SDEdit and Prompt-to-Prompt, our results achieve higher directional similarity for the same image similarity values, indicating it better performs the desired edit. These findings are measured on average across 2000 edits, and we further validate them using a different CLIP model in Fig. 23 of the supplement. Outperforming Prompt-to-Prompt may seem surprising, since it is used in our training data generation, and in Appendix B we discuss possible causes for this improvement.

4.2. Ablations

In Fig. 10, we provide quantitative ablations for both our choice of dataset size and our dataset filtering approach described in Section 3.1. Decreasing the size of the dataset typically results in decreased ability to perform more significant image edits, instead only performing subtle or stylistic image adjustments (and thus, maintaining a high image similarity score, but a low directional score). In contrast, removing the CLIP filtering from our dataset generation reduces the overall image consistency with the input image.

We also provide an analysis of the effect of our two classifier-free guidance scales in Figure 4. Increasing \( s_T \) results in a stronger edit applied to the image (i.e., the output agrees more with the instruction), and increasing \( s_I \) can help preserve the spatial structure of the input image (i.e., the output agrees more with the input image). We find that
values of $s_T$ in the range $5−10$ and values of $s_I$ in the range $1−1.5$ typically produce the best results. In practice, and for the results shown in the paper, we find it beneficial to adjust guidance weights for each example to get the best balance between consistency and edit strength.

5. Discussion

We demonstrate an approach that combines two large pretrained models, a large language model and a text-to-image model, to generate a dataset for training a diffusion model to follow written image editing instructions. While our method is able to produce a wide variety of compelling edits to images, including style, medium, and other contextual changes, there still remain a number of limitations.

Our model is limited by the visual quality of the generated dataset, and therefore by the diffusion model used to generate the imagery (in this case, Stable Diffusion [51]). Furthermore, our method’s ability to generalize to new edits and make correct associations between visual changes and text instructions is limited by the human-written instructions used to fine-tune GPT-3 [7], by the ability of GPT-3 to create instructions and modify captions, and by the ability of Prompt-to-Prompt [17] to modify generated images. In particular, our model struggles with counting numbers of objects and with spatial reasoning (e.g., “move it to the left of the image”, “swap their positions”, or “put two cups on the table and one on the chair”), just as in Stable Diffusion and Prompt-to-Prompt. Additionally, we find that performing many sequential edits sometimes causes accumulating artifacts. Examples of failures can be found in Figure 13. Furthermore, there are well-documented biases in the data and the pretrained models that our method is based upon. The edited images from our method may inherit these biases or introduce others (Fig. 14 in the supplement).

Aside from mitigating the above limitations, our work also opens up questions, such as: how to follow instructions for spatial reasoning, how to combine instructions with other conditioning modalities like user interaction, and how to evaluate instruction-based editing. Incorporating human feedback, such as with the use of reinforcement learning, is another important direction for future work and could improve alignment between our model and human intentions.

Acknowledgments We thank Ilija Radosavovic, William Peebles, Allan Jabri, Dave Epstein, Kfir Aberman, Amanda Buster, and David Salesin. Tim Brooks is funded by an NSF Graduate Research Fellowship. Additional funding provided by a research grant from SAP and a gift from Google.
References


[22] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 2


Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In IEEE International Conference on Computer Vision (ICCV), 2019. 2


Suman Ravuri and Oriol Vinyals. Classification accuracy score for conditional generative models. Advances in neural information processing systems, 32, 2019. 2

Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2287–2296, 2021. 2


Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo


