
Prompt-Based Localized Image Editing

Smit Patel

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
smotp@cs.cmu.edu

Mehul Jain

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
mehul.j2@cs.cmu.edu

Avi Sharma

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
avisharm@cs.cmu.edu

Nikhil Reddy

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
nthambal@cs.cmu.edu

Abstract

This 11-785 project explores methods to enhance prompt-based image editing capabilities of InstructPix2Pix, a deep learning pipeline that takes in an image and edit instruction to generate an edited version of that image. We will investigate and implement improvements to this baseline model by introducing localized image edits through segmentation techniques. These localized image edits will enable more fine-grained editing of images and thus improve generation.

1 Overview and Context

1.1 Motivation

The problem with traditional image editing tools is that it requires manual effort and domain expertise, which makes them more inaccessible for non-expert users. Generative models and other, more recent advances in deep learning have shown their ability for text-conditioned image editing, as well as their potential for easier, intuitive, and powerful editing.

Challenges in current systems include things like edit fidelity, prompt interpretability, and computational efficiency. The challenge with edit fidelity is that edits can often be imprecise or affect unintended regions. Prompt interpretability refers to the fact that current systems might misunderstand user instructions. And computational efficiency refers to the fact that current systems may have slow inference or high resource demands.

In order to make image editing more controllable, reliable, and efficient, we need to make image edits more granular. The applications of this work will be in content creation, design automation, marketing, entertainment, and personalized media.

1.2 Objective

The primary objective of this project is to analyze and refine an existing deep learning-based image editing model to achieve more reliable and context-aware edits. Some goals include:

- Run and evaluate baseline InstructPix2Pix model. Verify baseline metrics (CLIP Image Similarity and CLIP Text-Image Direction Similarity).

- Improve baseline metrics using segmentation techniques. Use masks to localize edits and preserve unchanged regions.
- Evaluate performance improvements, both quantitatively (CLIP metrics) as well as qualitatively (visual inspection).
- Assess generalization across domains (objects, scenes, etc.) and across types of edits (addition/removal, attribute change, etc.)

2 Related Works and Background

2.1 InstructPix2Pix

Existing editing approaches often require masks or semantic segmentations. They also depend on complex model inversion or optimization loops and cannot generalize to unseen edit instructions without retraining. The goal of the authors of InstructPix2Pix (Brooks et al., [2023]) was to create a general-purpose, user-friendly editing framework to execute natural language instructions on real images without masks, inversion, or per-example fine-tuning. The model adapts Stable Diffusion into a dual-conditional diffusion architecture, where the input image is treated as a conditional context and the text instruction is also encoded as a separate condition. This is done to balance edit fidelity with source preservation. The strengths of this methodology include faster inference, good generalization to unseen instructions, and good balance of edit realism and source preservation. This work forms the baseline system under evaluation and provides the core editing mechanism that we aim to improve using segmentation-based guidance.

2.2 Segmentation

Text-to-image models might overdo or incorrectly edit an image when only local changes are desired. Sometimes, even in spite of giving a prompt-based instruction for a specific local edit, global alterations can take place. An Item is Worth a Prompt: Versatile Image Editing with Disentangled Control (Feng et al., [2025]) looks to provide region-aware, prompt-disentangled control for fine-grained edits. The researchers created this model to decompose the prompt-image interaction. Instead of one prompt for the whole image, the model would learn a region-specific embedding tied to a localized prompt (i.e. "hat," "shirt," etc.). The edits are applied through group cross-attention layers that condition the diffusion model on these item-prompt pairs. The model was based on stable diffusion and utilizes per-region latent embeddings, cross attention layers, and modular editing logic for localized edits. This enables fine-grained edits over the image while preserving the background and unrelated areas in the image. This relates to our work since it pertains to our work around segmenting the image and utilizing region-aware editing in our pipeline.

3 Methodology

3.1 Model Description

Our approach enhances instruction-based image editing by introducing localized editing mechanisms to improve object isolation on a 512x512x3 (HxWxC) image. The current InstructPix2Pix model applies transformations globally, often affecting unintended areas. To address this, we propose the integration of an object localization module that enables fine-grained modifications to targeted objects while preserving the surrounding image content. To explain our methodology, we'll use the example prompt "Change the temple to a castle" on the below image:



Figure 1: Original Image

The first step in this process is to extract the subject from the prompt. For example, in the prompt "Change the temple to a castle," the subject is "the temple". We use GPT-4o on the input prompts and ask it to return the subject of the prompt. For cases where the model cannot find a subject, we instruct it to return "None". For example, if the prompt is "Make it blue", it is unclear what the subject is, and thus the LLM returns "None." In these cases, we skip our pipeline and resort to the baseline approach.

The next step in the process is to locate the subject in the original image with a bounding box. For object localization, we use GroundingDINO (Liu et al., 2024). It is a state-of-the-art vision-language model that combines text and visual input to perform open-set object detection and localization with unprecedented accuracy. It employs a dual encoder transformer-based architecture with a novel cross-modal attention mechanism to establish fine-grained correspondences between textual descriptions and image regions at multiple scales. The model processes arbitrary text prompts alongside hierarchical visual features extracted from convolutional and transformer pathways to generate precise bounding boxes with confidence scores. GroundingDINO achieves remarkable zero-shot transferability by leveraging language priors from large-scale text encoders (BERT) and visual knowledge from vision transformers (ViT), enabling detection of novel object categories not encountered during training. Running GroundingDINO on Figure 2 with the subject "the temple" gives us the following 512x512x3 image with a bounding box:



Figure 2: Original Image with Bounding Box

Post objection localization, we use Segment Anything Model (SAM) Kirillov et al., 2023, which is a foundation model for image segmentation designed to produce high-quality object masks from various input prompts with remarkable adaptability across domains and tasks. SAM employs a sophisticated three-component architecture consisting of an image encoder based on a modified Vision Transformer (ViT-H) that processes the input image once to create an image embedding,

a flexible prompt encoder that can handle different prompt types (points, boxes, masks, or text), and a mask decoder that efficiently transforms these embeddings into high-resolution segmentation masks. The model was trained on Meta’s SA-1B dataset containing over 1 billion masks across 11 million diverse images, establishing an unprecedented scale for segmentation model pretraining. SAM demonstrates exceptional zero-shot generalization capabilities, maintaining high-quality mask prediction even for object categories, domains, and tasks not encountered during training. Running SAM on the GroundingDINO output gives us two 512x512x3 outputs: (1) segmentation of the subject in the original image with the background blacked out, (2) an inverse mask where the background is white and the segmented region is blacked out. The two outputs are shown below:



Figure 3: Segmented Image

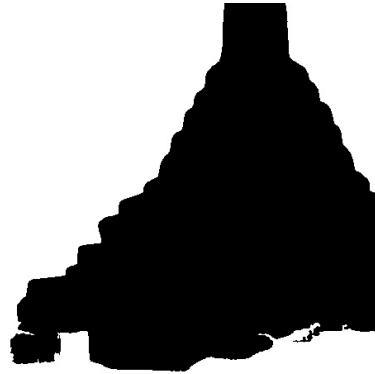


Figure 4: Inverse Mask

Finally, the segmented image and the original prompt are given to InstructPix2Pix so that it can perform the needed edit on the segmented image. We then perform a bitwise AND operation between the original image (Figure 2) and the inverted mask (Figure 5) to isolate the background in the original image. As the last step, we perform a bitwise OR operation between the isolated background and the InstructPix2Pix output to get the final 512x512x3 image:



Figure 5: Final Output

The entire pipeline can be summarized as follows:

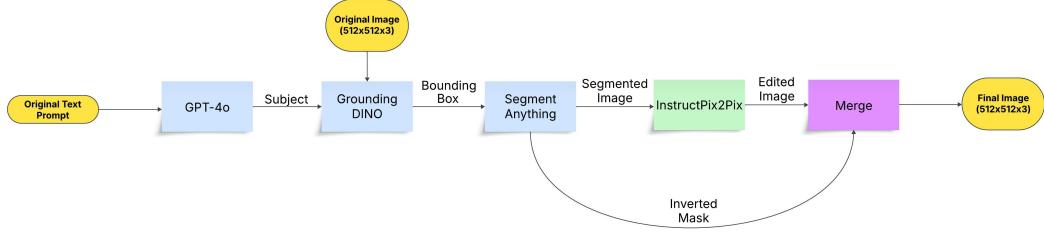


Figure 6: Methodical overview of our image-segmentation and masking approach

The parameter counts of the models that make up the pipeline are below:

Model	Parameters
sam-vit-base	93.7M
grounding-dino-base	233M
InstructPix2Pix	~1.1B

Table 1: Comparison of model sizes by parameter count

3.2 Dataset

To ensure consistency with previous work and improve generalization, we use the InstructPix2Pix dataset (“InstructPix2Pix Dataset”, n.d.). The authors provided a script to download the entire dataset. Since the original is almost 500 GB and thus too large to run on, we edited this script to import a portion of the data totaling 14 GB. Based on our observations, this portion seems to be a representative sample of the original dataset.

To perform inference on this dataset, we wrote a script to go through each directory of the dataset and extract pairs of input image and the associated edit prompt. Then we run each pair through our pipeline to generate the output image.

3.3 Evaluation Metric

We use two evaluation metrics. The first metric is CLIP Image Similarity. This measures how much the edited image agrees with the input image. The goal is to have high similarity to ensure that the edited image does not deviate significantly from the input image.

The second evaluation metric is CLIP Text-Image Direction Similarity. This measures how much the change from the input image to the output image agrees with the change in text captions for both images. The goal is also to have high similarity to ensure that the pipeline is performing the edit correctly.

Both metrics rely on computing cosine similarity in the latent space. Images are converted into a vector embedding using an image encoder while text is converted into a vector embedding using a text encoder. For the first metric, the two images are converted into vector embeddings and then the cosine similarity is taken between them. For the second metric, the two images and two text captions are converted into their vector embeddings. Then the difference in vector embeddings is computed for images and text respectively. Finally, cosine similarity is calculated between the difference in image embeddings and difference in text embeddings. The formula for cosine similarity is shown below:

$$\cos(\theta) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|} \quad (1)$$

4 Baselines and Extensions

4.1 Baseline Selection

Our baseline is **InstructPix2Pix** Brooks et al., [2023], a foundational model for instruction-driven image editing. InstructPix2Pix is a strong baseline for our project because it directly addresses the task of prompt-based image editing, modifying real images based on natural language instructions without requiring masks or fine-tuning. It is a widely recognized model in the literature, with clearly defined evaluation metrics (CLIP Image Similarity and CLIP Text-Image Direction Similarity) that allow for systematic comparison. Its design highlights a key limitation: global edits that unintentionally alter irrelevant regions, which directly motivates our proposed segmentation-based extension. Thus, it provides both a relevant starting point and a clear benchmark for improvement.

We draw further inspiration from **An Item is Worth a Prompt** Feng et al., [2024] where the authors propose disentangling image-prompt associations to achieve region-specific edits without affecting the global image context. Their pipeline employs GroundedSAM Ren et al., [2024] a two-stage framework that first localizes objects based on text prompts using GroundingDINO and then refines segmentation masks with the Segment Anything Model (SAM), enabling highly precise and prompt-aligned local editing. Following this principle, we introduce segmentation-based extensions to the InstructPix2Pix pipeline, aiming for more controlled and precise edits.

Moreover, baseline performance was systematically studied before proposing extensions, ensuring that limitations and failure modes of the original model were well-understood. While InstructPix2Pix is effective at following global edit instructions, it often struggles with localized edits where only specific regions of the image should be modified. Because the model applies transformations across the entire image, unintended changes to background or unrelated objects are common, leading to a loss of fine-grained control and reduced edit fidelity.

4.2 Extensions and Experiments

Our primary extension involves a plug-and-play segmentation module, integrating a two-stage pipeline of object localization and mask generation to address the baseline’s limitation of making localized edits:

1. **GroundingDINO** Liu et al., [2024] is used to detect objects corresponding to the prompt and output bounding boxes.
2. **Segment Anything Model (SAM)** Kirillov et al., [2023] is applied to generate precise masks from these bounding boxes.

The masked image, retaining only the segmented object, is then fed into InstructPix2Pix for localized editing. We performed systematic ablation studies to evaluate different design choices and failure cases.

4.2.1 Masking Variants: White vs Black Background

We experimented with two types of background masking after segmentation: white-masked backgrounds, where non-object regions are set to a pixel value of 255, and black-masked backgrounds, where non-object regions are set to a pixel value of 0. Our experiments showed that black-masked images maintained editing fidelity better, possibly because black backgrounds introduce less visual bias during model inference compared to white patches, which can sometimes interfere with the model’s interpretation of the scene.

4.2.2 Model Size Ablations: DINO-Base vs DINO-Tiny

We compared the performance of **GroundingDINO-base** and **GroundingDINO-tiny** models. DINO-base reliably detected objects across a wide variety of images, demonstrating strong generalization even in challenging scenarios. In contrast, DINO-tiny struggled particularly when the object occupied over 84% of the image or when the object blended smoothly into the background, leading to missed or inaccurate detections. These observations highlighted the necessity of using high-quality object detection models to ensure robust and precise localized editing.

4.2.3 Prompt Subject Extraction: NER Failures and LLM-based Approach

We initially experimented with extracting object labels from prompts using traditional Named Entity Recognition (NER) models, but these approaches often failed on prompts like "put in a lake" or "add a watermelon," where the subject is generic or not captured by standard entity types. NER systems are typically optimized for well-defined categories like persons or locations and lack the semantic flexibility to interpret implied objects in imperative or creative language. To address this, we switched to an LLM-based subject extraction method using carefully crafted prompts to interpret the intended object. For ambiguous or incomplete prompts, the system returns "None", indicating that the baseline should instead be run on that example.

4.2.4 Object Detection vs Direct Prompt Segmentation

Inspired by ablations in object detection models, we considered whether direct prompt-to-segmentation approaches could replace explicit detection. However, direct segmentation often produced imprecise masks. Moreover, Intermediate object localization (GroundingDINO) significantly improved segmentation quality. This validated our modular pipeline approach, emphasizing the need for clear object grounding before segmentation.

4.3 Key Takeaways from Ablations

- Masking choice (white vs black) affects model behavior; black masks are more robust.
- Larger detection models (DINO-base) generalize better across complex images.
- NER-based prompt subject extraction is not robust and hence LLM employment is preferred.
- Explicit object localization before segmentation is crucial for high-quality editing.

These systematic evaluations informed our design decisions and strengthened the proposed extension.

5 Results & Analysis

5.1 Results

In our experiments, we evaluated two key variations of our localized image editing approach against the baseline InstructPix2Pix model. We evaluated the results using CLIP Image Similarity and CLIP Direction Similarity metrics across our test dataset to quantify the effectiveness of our proposed methods.

We also have plots for the CLIP scores. The graphs show the Delta CLIP Similarity values plotted against Sample Index for 183 samples from our test set. The y-axis represents the change in similarity scores, with positive values indicating improvement and negative values indicating degradation. The blue line shows how the similarity scores changed for each input sample, and the dashed gray line at $y=0$ represents no change.

5.1.1 Full Pipeline (GroundingDINO + SAM)

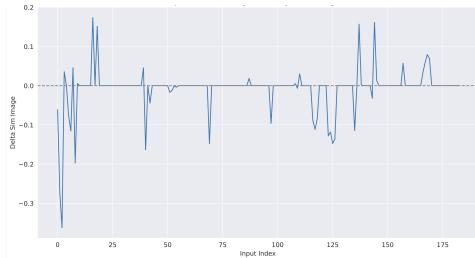


Figure 7: Image Similarity Plot

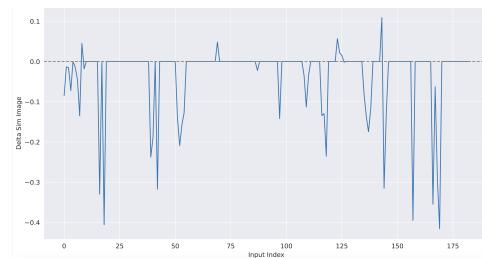


Figure 8: Directional Similarity Plot

Metric	Value
Total Samples	183
Average Improvement	-0.0076
Median Improvement	0.0000
Std Dev of Improvement	0.0553
Minimum Delta	-0.3622
Maximum Delta	0.1736
Samples Improved	18/183 (9.84%)
Samples Degraded	25/183 (13.66%)

Table 2: Image Similarity Results

Metric	Value
Total Samples	183
Average Improvement	-0.0303
Median Improvement	0.0000
Std Dev of Improvement	0.0847
Minimum Delta	-0.4159
Maximum Delta	0.1089
Samples Improved	6/183 (3.28%)
Samples Degraded	37/183 (20.22%)

Table 3: Direction Similarity Results

5.1.2 GroundingDINO Box-Only Segmentation

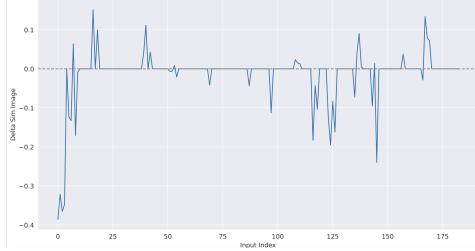


Figure 9: Image Similarity Plot

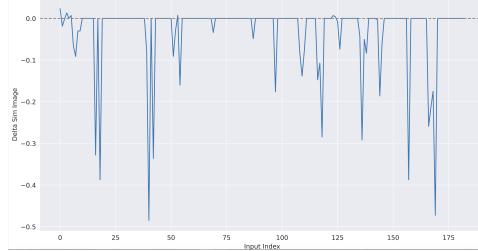


Figure 10: Directional Similarity Plot

Metric	Value
Total Samples	183
Average Improvement	-0.0130
Median Improvement	0.0000
Std Dev of Improvement	0.0683
Minimum Delta	-0.3857
Maximum Delta	0.1510
Samples Improved	18/183 (9.84%)
Samples Degraded	25/183 (13.66%)

Table 4: Image Similarity Results

Metric	Value
Total Samples	183
Average Improvement	-0.0299
Median Improvement	0.0000
Std Dev of Improvement	0.0855
Minimum Delta	-0.4848
Maximum Delta	0.0234
Samples Improved	6/183 (3.28%)
Samples Degraded	37/183 (20.22%)

Table 5: Directional Similarity Results

5.2 Analysis

The results from our experiments suggest several key insights about the challenges of applying segmentation techniques to prompt-based image editing:

1. While segmentation should theoretically allow for more precise edits, our results indicate that isolating objects with GroundingDINO and SAM may sometimes disrupt the holistic understanding that InstructPix2Pix leverages when processing the entire image
2. By segmenting objects and processing them separately, our pipeline may lose important contextual information that helps the baseline model determine appropriate edits. This context loss appears to outweigh the benefits of localized editing in many cases
3. GPT-4o's extraction of subjects from prompts worked in many cases but still struggles in certain cases. The object detection model also fails to identify the subjects accurately in some cases leading to incorrect segmentation
4. Despite the average negative impact, the maximum positive deltas (0.1736 for image similarity in the full pipeline and 0.1510 in the GroundingDINO-only approach) suggest that our

approach does significantly improve results in specific scenarios. Identifying these favorable conditions could lead to a more targeted application of our pipeline.

The comparable performance metrics between our full pipeline and the GroundingDINO-only approach suggest that the primary challenge may not be the precision of segmentation but rather the fundamental approach of isolating objects before editing. The added complexity of SAM did not substantially change the outcomes.

6 Future Directions

Building on the insights gained from our explorations of localized image editing, we propose the following future directions:

- **Contextual Preservation:** Explore methods to maintain contextual information when performing localized edits. This might involve encoding background context as conditioning information even when editing isolated objects.
- **User-Guided Segmentation:** Incorporate interactive feedback to allow users to correct or refine automatic segmentations when needed. This would enable more precise control in edge cases where automatic detection fails or is ambiguous.
- **Spatially Awareness:** Modify InstructPix2Pix’s architecture to incorporate a spatial semantic alignment module that generates attention maps directly from text prompts possibly enabling object-focused editing without external segmentation while preserving contextual understanding.

7 Conclusion

Although our localized image editing approach shows promise by showing significant improvements in several examples, our quantitative evaluation revealed challenges with our pipeline. We found that segmenting objects sometimes disrupted the contextual understanding that InstructPix2Pix leverages, leading to reduced performance in many cases. Despite these challenges, the positive results in select examples suggest potential for targeted applications. Moving forward, we aim to explore methods for contextual preservation during localized editing, user-guided segmentation refinement, and modifications to the original InstructPix2Pix architecture to enable object-focused editing while maintaining contextual awareness. Though our approach did not consistently improve the baseline, we’ve developed valuable insights into the complex balance between localized control and contextual coherence in prompt-based image editing.