Appendix

A. More Details on Dataset Filtering

In order to decide the candidate objects to be potentially edited, we first check if the object is sufficiently large. This is because the object needs to be at least distinguishable in the image and to have enough pixels to make any meaningful edits. For this reason, we filter out objects whose bounding box size is smaller than 0.5% of the whole image. Also, in order to avoid a situation that a query asks to edit the unseen area of an object, we require the whole body of the target object to be shown in the image; so we eliminate objects that have bounding box adjacent to the edge of the image, regarding that those objects may have been cropped. We further eliminate occluded objects with help from a pre-trained Vision Question and Answering (VQA) model [17]. Given an image with an object A, we ask the VQA model to answer the following six yes-or-no questions, and only the objects that receive 4 or more desired answers are kept.

- *Is the A hidden behind another object?*
- *Is part of the A covered by another object?*
- *Is the A partially outside the image frame?*
- *Is part of the A blocked by something else in the scene?*
- *Are parts of the A visible?*
- *Is the A fully in view without anything blocking it?*

Lastly, an off-the-shelf instance segmentation model should be able to detect the objects to automate the evaluation process using our approach. Thus, we filter out objects that have its IoU lower than 0.5 between the annotated bounding box in GQA dataset and the detected segment.

B. More Details on Editing Query

B.1. Details on Generation Rules

Object-centric Queries. For OBJECT ADDITION, the target object class $c_{o'}$ and its desired placement $r_{o'}$ is required. To select feasible target object type $c_{o'}$, we conduct a statistical analysis of the relative position information from the base dataset to identify relevant objects that frequently have relationships with the anchor object o in the specified positions. To decide all feasible location of target object o' relative to o, we utilize again the relative position information annotated in the base dataset to accurately determine a placement that accounts for depth. Additionally, in order to make the addition feasible, we restrict the target location to some space that is unoccupied by other objects in the image based on scene graph annotation in the base dataset. We also make sure if there is large enough margin in the image to generate any additional object. These ensure that we generate a query only with a plausible target object at a feasible location. We also make sure not to ask the model to generate any object class that already exists in the image. This is to avoid having two or more same class objects in the image that might cause confusion during the segmentation process at evaluation.

The OBJECT REMOVAL task involves removing an editable object from an image. There is little feasibility restriction for this task, as long as the target object o has been detected and belongs to one of the covered classes in $\mathcal{C}$. While this task appears simple, removing the main object from the image can significantly impact its description. As maintaining the original image description is crucial for editing consistency, we exclude object removal tasks when we evaluate description-based methods.

The OBJECT REPLACEMENT task involves altering the core identity of an object, keeping the original position unchanged. The replacement target class $c_{o'}$ is selected similarly to the OBJECT ADDITION task, by choosing an object from the dataset that has a realistic and contextually appropriate relationship with all other objects in the image. This ensures that the edited image maintains plausible object positioning. Again, we avoid any object class that already exists in the image to prevent confusion at evaluation.

For OBJECT RESIZING, when the target object is too small (or too large) relative to the image size, we only generate a query to make the object larger (smaller).

For OBJECT ATTRIBUTE CHANGE, it is crucial to preserve core identity of the object o while editing the specific characteristics mentioned in the query. We similarly curate a list of attribute words from the base dataset. As we do not know what kind of attributes an object can have other than the annotated ones, we restrict changes only to the annotated attributes. In addition, when choosing a replacement attribute, the option set is structured slightly different for each attribute type (color, state, material, action). A typical option set of certain object class will look like as follows:

- Color: ‘black’, ‘green’, ‘brown’, ...

- State: ‘wet’, ‘dry’, ‘new’, ‘old’, ‘rusted’, ...
- Material: ‘wood’, ‘metal’, ...
- Action: ‘standing’, ‘running’, ...

If we are to change an attribute other than those in the State group, we simply choose a replacement from the same attribute group (*i.e.*, ‘green’ from Color to replace ‘black’). On the other hand, attributes in the State group usually cannot be replaced with any random state attributes. For example, a state ‘wet’ should not be replaced with ‘rusted’. Thus, we define feasible alternatives for each state individually, based on the statistics from the base dataset annotations (*i.e.*, couple ‘wet’ and ‘dry’).

Non-object-centric Queries. The full list of elements on BACKGROUND CHANGE and STYLE CHANGE options are as follows:

$\mathcal{B}$ = {‘beach’, ‘pine forest’, ‘urban city’, ‘desert’, ‘snow field’, ‘country side farm’, ‘tropical jungle’, ‘vineyard’, ‘lake side’, ‘mountain top’, ‘living room’, ‘cave’, ‘art gallery’, ‘ancient ruins’, ‘space station’, ‘grass field’, ‘train station’, ‘library’, ‘restaurant’, ‘airport’, ‘hospital’, ‘gym’, ‘zoo’, ‘aquarium’, ‘museum’, ‘concert hall’, ‘stadium’},

$\mathcal{S}$ = {‘watercolor painting’, ‘Van Gogh art’, ‘oil painting’, ‘cartoon’, ‘gray scale’, ‘pencil sketch’, ‘mosaic art’, ‘pop art’, ‘graffiti art’, ‘ancient Egyptian art’}.

To prevent editing the background to the one that is already the original background, we select target background (b) from half of the options with the lowest CLIP alignment with the original image (I_0).

B.2. Detailed Flow for Edit Description and Instruction Generation

For description-based editing, we first generate a caption for the original image (C_0). As the base dataset does not provide a natural language description of the images, we use Llama3 [8] to generate C_0. To generate a caption focusing on the editable objects, we directly demand it to use the exact object names and repeat generation until the caption contains the exact names. We also require the model to answer within 60 words to avoid long captions beyond the maximum input length of the editing models.

Then, we refine the generated captions to insert available attributes of the editable objects using GPT4 [33]. Specifically, we ask it to first remove any attribute descriptions of the object in the caption to prevent collision, and then to insert desired attributes of the objects using the exact words we provide; *e.g.*, “a photo of a crimson cat” (first draft) $\rightarrow$ “a photo of a cat” (original attributes removed) $\rightarrow$ “a photo of a wet red cat” (desired attributes inserted, C_0).

We generate a target caption (C_e) specific to each task. For OBJECT REPLACEMENT queries, we ask GPT to find and replace the object with the desired one. Then, we ask it to correct any grammatical errors (*e.g.*, “a photo of a person with his cat” (C_0) $\rightarrow$ “a photo of a phone with his cat” (original object replaced) $\rightarrow$ “a photo of a phone with its cat” (grammar fixed, C_e)). For OBJECT ADDITION, OBJECT RESIZING, and BACKGROUND CHANGE, we ask GPT to add a desired information to the original caption. For OBJECT ATTRIBUTE CHANGE, we simply replace the original attribute word with a desired one. For STYLE CHANGE queries, we attach a short prefix sentence describing about the style of the image, and ask GPT to combine the two sentences into one; *e.g.*, “A cat is sitting on a couch.” (C_0) $\rightarrow$ “An oil painting art. The art contains following: A cat is sitting on a couch.” (prefix attached) $\rightarrow$ “An oil painting art of cat sitting on a couch.” (Combined sentence, C_e).

To finalize, we test run generated original and target captions (I_0 and I_e), through the text encoders of the editing models to check their compatibility. We manually fix captions with any problems oddness. The full list of LLM prompts used to generate captions are provided in Appendix B.3.1.

For instruction-based editing, we manually craft the instruction templates for each edit type, *e.g.*, “change the color of the $\{object\}$ from $\{a\}$ to $\{b\}$”, and fill them with desired words to finalize. The full list of templates are provided in Appendix B.3.2.

B.3. Full List of LLM Prompts and Instruction Templates

Below, we provide the full list of LLM prompts and templates we used to generate each caption and instructions.

B.3.1 LLM Prompts

For original caption C_0 generation:

```
C = Llama3('Describe the photo focused on the details of the {editable object} in
            single sentence. You must use the exact word "{editable object}" to
```

```
            refer to the {editable object}. The description must be within 60
            words. Print only the generated description.')

C_1 = GPT4('You are an editor. You need to replace nouns with meaning strictly
            identical to {editable object} with "{editable object}" in the
            following sentence: "{C}". Do not change any pronouns. Do not change
            any other words. Print only the edited sentence.')

C_2 = GPT4('You are an editor. You need to remove modifiers describing the
            {editable object} in the following sentence: "{C1}". Do not remove
            any other modifiers that is not describing the {editable object}. Do
            not change any other words. Print only the edited sentence.')

C_0 = GPT4('You are an editor. You need to add following modifiers: "{attributes}"
            to {editable object} in the following sentence: "{C_2}". Do not change
            any other part of the sentence. Print only the edited sentence.')
```

For target caption C_e generation for OBJECT ADDITION:

```
C_e = GPT4('You are an editor. You need to add information of an additional
            subject "{target object}" into the following sentence: "{C_0}". The
            {target object} is {location} the {reference object}. You must include
            information of location of the {target object}. Do not change any
            other original inforamtion. Print only the edited sentence.')
```

For target caption C_e generation for OBJECT REPLACEMENT:

```
C_1 = GPT4('You are an editor. You need to remove any phrase or clause describing
            about the {original object} in the following sentence: "{C_0}". Do not
            remove {original object} it self. Do not remove any other information.
            Print only the edited sentence.')

C_2 = GPT4('You are an editor. You need to correct unnatural pronouns of following
            sentence: "{C_1}", if there is any. Do not change the word
            "{target object}". Do not change any other words. Print only the
            edited sentence.')

C_e = GPT4('You are an editor. You need to replace {original object} with
            {target object} in the following sentence: "{C_2}". Do not change any
            other information. Print only the edited sentence.')
```

For target caption C_e generation for OBJECT RESIZING:

```
C_e = GPT4('You are an editor. You need to add size information of the
            {editable object} in the following sentence: "{C_0}". The size of the
            {editable object} is {size}. Do not chnage any other part of the
            sentence. Print only the edited sentence.')
```

For target caption C_e generation for OBJECT ATTRIBUTE CHANGE:

```
C_e = C_0.replace({original attribute}, {target attribute})
```

For target caption C_e generation for BACKGROUND CHANGE:

```
C_e = GPT4('You are an editor. You need to add or change background information
            into the following sentence: "{C_0}". The desired background is
            {target background}. Do not change any other original inforamtion.
            Print only the edited sentence.')
```

For target caption C_e generation for STYLE CHANGE:

```
C_1 = '{target style} style image. The image contains: ' + C_0
C_e = GPT4('You are an editor. You need combine following sentences: "{C_1}" into
            one single sentence. Print only the edited sentence.')
```

B.3.2 List of Caption Generation Prompts

For instruction C generation for OBJECT ADDITION:

```
# e.g., "add a chair next to the person"
C = "add {a/an} {target object} {relation} the {reference object}."
```

For instruction C generation for OBJECT REMOVAL:

```
# e.g., "remove the chair from the image"
C = "remove the {target object} from the image."
```

For instruction C generation for OBJECT REPLACEMENT:

```
# e.g., "replace the chair with a person"
C = "replace the {original object} with {a/an} {target object}"
```

For instruction C generation for OBJECT RESIZING:

```
# e.g., "make the chair smaller"
C = "make the {target object} {larger/smaller}"
```

For instruction C generation for OBJECT ATTRIBUTE CHANGE:

```
# type in ["color", "matrial", "state", "action"]
# e.g., "change the color of the car from black to white"
C = "change the {type} of the {target object} from {before} to {after}"
```

For instruction C generation for BACKGROUND CHANGE:

```
# e.g., "change the background to pine forest"
C = "change the background to {target background}"
```

For instruction C generation for STYLE CHANGE:

```
# e.g., "change image style to oil painting style"
C = "change image style to {target style} style"
```

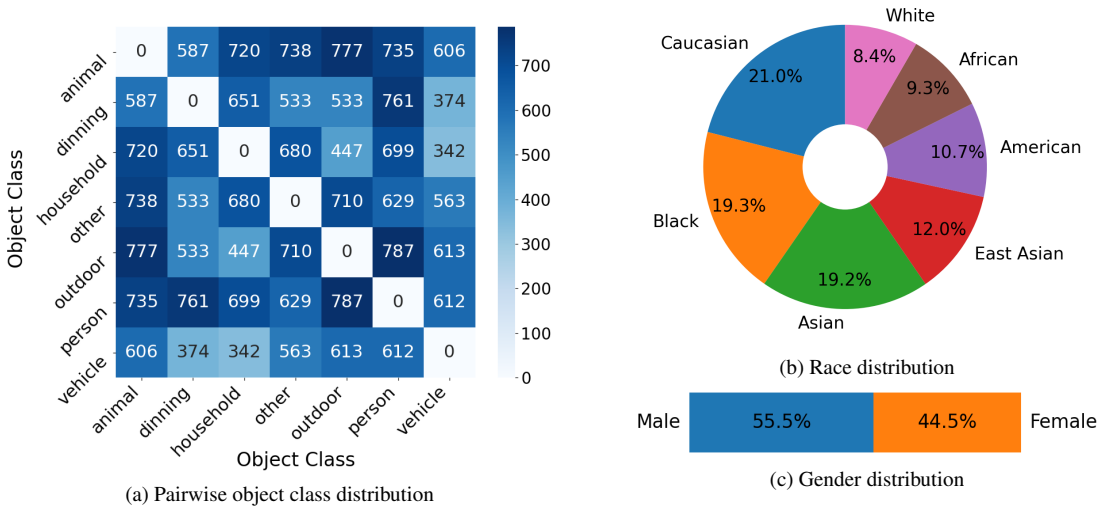
B.4. Balanced Query Types and Options

To mitigate bias in edit queries, we extracted valid relationships from large-scale dataset while ensuring a balanced distribution of object types and relations. The pairwise object type counts in Fig. Ia and Fig. 4 indicate that our dataset and queries well balanced throughout object classes. Additionally, we assess potential gender and racial biases. Since direct labels are unavailable, we estimate the most probable class by computing CLIP similarity scores between 'person' instances and various gender- and race-related terms. The results presented in Fig. Ib and Fig. Ic suggest that our dataset and queries retains balance between gender and racial types.

C. More Details on Evaluation Pipeline

C.1. Task-specific Evaluation Workflow

We provide detailed evaluation workflow described in Fig. 5 and Fig. XI.



(a) Pairwise object class distribution

(b) Race distribution

(c) Gender distribution

Figure I. **Data distribution in editing queries.** (a) Pairwise object class counts (b) Race distribution within the *person* class. (c) Gender distribution within the *person* class.

C.1.1 Object Addition

For a satisfactory Object Addition, the newly generated object o' should be 1) within the desired class ($c_{o'}$) and 2) located at the desired location ($r_{o'}$), while 3) retaining the background. The first point is quantified by σ^{OF}_{det} and $\sigma^{OF}_{\text{clip,c}}$, where $\sigma^{OF}_{\text{clip,c}}$ denotes $\sigma^{OF}_{\text{clip}}$ computed between the generated object segment M_o and the object class name o (*e.g.*, CLIP alignment between segmented pizza and word "pizza" in Fig. 5a). To quantify the second point, we first square crop the edited image (I_e) to be minimal in size while enclosing both the reference object (o) and the generated one o' (*e.g.*, cropped image containing pizza and microwave in Fig. 5a). Then, we compute the CLIP alignment score between the cropped image and segment of instruction (C) that describes $r_{o'}$ (*e.g.*, "pizza under microwave" in Fig. 5a). We define this as $\sigma^{OF}_{\text{clip,r}}$. These two points make up σ^{OF} as follows:

$$\sigma^{OF} = \frac{w^{OF}_{\text{clip,c}}\sigma^{OF}_{\text{clip,c}} + w^{OF}_{\text{clip,r}}\sigma^{OF}_{\text{clip,r}} + w^{OF}_{\text{det}}\sigma^{OF}_{\text{det}}}{w^{OF}_{\text{clip,c}} + w^{OF}_{\text{clip,r}} + w^{OF}_{\text{det}}}. \tag{6}$$

The third point is evaluated with σ^{BC}. The original and edited background for σ^{BC} are obtained by masking the o' region from both original (I_0) and edited (I_e) image. We mask out o' region from I_0 as well (although it may not contain any meaningful features) to remove any discrepancy due to the masked region in the edited background. When o' is not detected from I_e, σ^{OF} is set to 0, and σ^{BC} is computed with the entire I_0 and I_e. To sum up, the Object Addition task is evaluated with two scores, the Object Fidelity σ^{OF} and Background Consistency σ^{BC}.

C.1.2 Object Removal

For Object Removal task to be successful, target object o should be 1) undetected from the edited image (I_e), while 2) retaining the background. The first point is quantified by $1 - \sigma^{OF}_{\text{det}}$ and $1 - \sigma^{OF}_{\text{clip,c}}$, explained in Appendix C.1.1. We modify the scores to $1 - \sigma^{OF}_{*}$ format to suit the task's character (the more unlike the edited object is to the original object class, the better the edit is). These two scores make up σ^{OF} as follows:

$$\sigma^{OF} = \frac{w^{OF}_{\text{clip,c}}(1 - \sigma^{OF}_{\text{clip,c}}) + w^{OF}_{\text{det}}(1 - \sigma^{OF}_{\text{det}})}{w^{OF}_{\text{clip,c}} + w^{OF}_{\text{det}}}. \tag{7}$$

The second point is evaluated with σ^{BC}. The original and edited background are obtained by masking the union of original and edited region of the object o from I_0 and I_e, respectively. We mask out the same region form I_0 and I_e for the same reason in Appendix C.1.1. When o is not detected from I_e, σ^{OF} is set to 1, and σ^{BC} is computed with backgrounds with only the original object regions masked. To sum up, the Object Removal task is evaluated with two scores, the Object Fidelity σ^{OF} and Background Consistency σ^{BC}.

C.1.3 OBJECT REPLACEMENT

For a successful OBJECT REPLACEMENT, the target object (o) should be replaced with a new object (o') 1) within desired class ($c_{o'}$), while 2) retaining the location of o, 3) without affecting the background. The first point is quantified by σ^{OF}_{det} and $\sigma^{OF}_{\text{clip,c}}$ explained in Appendix C.1.1, which makes up σ^{OF} as:

$$\sigma^{OF} = \frac{w^{OF}_{\text{clip,c}}\sigma^{OF}_{\text{clip,c}} + w^{OF}_{\text{det}}\sigma^{OF}_{\text{det}}}{w^{OF}_{\text{clip,c}} + w^{OF}_{\text{det}}}. \tag{8}$$

The second point is quantified by σ^{OC}_{pos}, and solely makes up $\sigma^{OC} = \sigma^{OC}_{\text{pos}}$. The third point is again quantified with σ^{BC}. The original and edited backgrounds are obtained by masking the union of o and o' region from I_0 and I_e. We mask out the same region form I_0 and I_e for the same reason in Appendix C.1.1. When o' is not detected from I_e, σ^{OF} and σ^{OC} is set to 0, and σ^{BC} is computed with backgrounds with only the original object regions masked. In sum, the OBJECT REPLACEMENT score is composed of three scores, Object Fidelity σ^{OF}, Object Consistency σ^{OC} and Background Consistency σ^{BC}.

C.1.4 OBJECT RESIZING

For OBJECT RESIZING, we define four conditions to satisfy. The target object o should be 1) correctly resized, retaining its 2) shape and 3) position, 4) without affecting the background. The first point is quantified by $\sigma^{OF}_{\text{size}}$, which solely makes up $\sigma^{OF} = \sigma^{OF}_{\text{size}}$. The second point is quantified by $\sigma^{OC}_{\text{lpips}}$, $\sigma^{OC}_{\text{dino}}$, and $\sigma^{OC}_{\ell 2}$, and the third point measured by σ^{OC}_{pos}. These make up σ^{OC} as follows:

$$\sigma^{OC} = \frac{w^{OC}_{\text{lpips}}\sigma^{OC}_{\text{lpips}} + w^{OC}_{\text{dino}}\sigma^{OC}_{\text{dino}} + w^{OC}_{\ell 2}\sigma^{OC}_{\ell 2} + w^{OC}_{\text{pos}}\sigma^{OC}_{\text{pos}}}{w^{OC}_{\text{lpips}} + w^{OC}_{\text{dino}} + w^{OC}_{\ell 2} + w^{OC}_{\text{pos}}}. \tag{9}$$

The fourth point is evaluated again with σ^{BC}. The original and edited background are obtained by masking the union of original and resized region of the object o from I_0 and I_e, respectively. We mask out the same region form I_0 and I_e for the same reason as in Appendix C.1.1. When o is not detected from I_e, σ^{OF} and σ^{OC} is set to 0, and σ^{BC} is computed with backgrounds with only the original object regions masked. In sum, the OBJECT RESIZING task is measured with three scores, Object Fidelity σ^{OF}, Object Consistency σ^{OC} and Background Consistency σ^{BC}.

C.1.5 OBJECT ATTRIBUTE CHANGE

We define five points to measure for the OBJECT ATTRIBUTE CHANGE task: 1) whether object's (o) attribute (a_i) is changed to the desired attribute (a_j), 2) retaining fundamental morphological characteristics, 3) position and 4) size of o, 5) without affecting the background. The first point is quantified by $\sigma^{OF}_{\text{clip,a}}$, which is $\sigma^{OF}_{\text{clip}}$ computed between the edited object segment M_o and the word of o and a combined (*e.g.*, "cream motorcycle " in Fig. 5e). $\sigma^{OF}_{\text{clip,a}}$ solely makes up $\sigma^{OF} = \sigma^{OF}_{\text{clip,a}}$

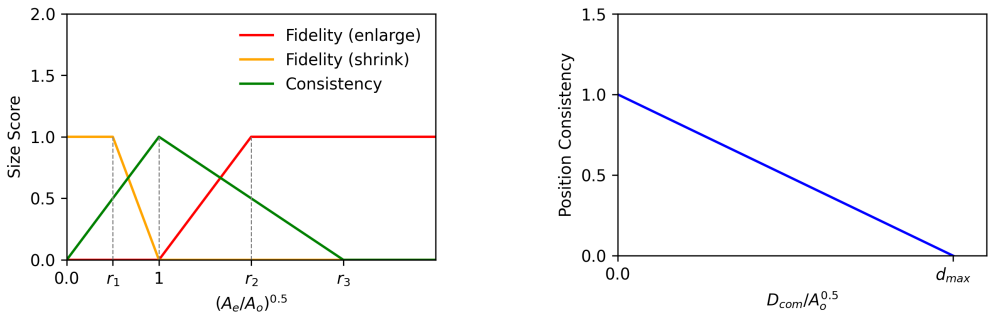
To quantify the second point, we dull out the details of object segments from I_0 and I_e by degradation (gray scaling and down-scaling) and Canny-edge detection [4]. This is to remove any inconsistencies due to the attribute edit and remain only the fundamental morphological characters of the object. Then, we compute σ^{OC}_{deg} and $\sigma^{OC}_{\text{edge}}$, which are σ^{OC} in Eq. (3) computed with the degraded object segment and its detected edges, respectively. The third and fourth points are quantified with σ^{OC}_{pos} and $\sigma^{OC}_{\text{size}}$. These three key-points make up σ^{OC} as:

$$\sigma^{OC} = w^{OC}_{\text{deg}}\sigma^{OC}_{\text{deg}} + w^{OC}_{\text{edge}}\sigma^{OC}_{\text{edge}} + w'^{OC}_{\text{pos}}\sigma^{OC}_{\text{pos}} + w'^{OC}_{\text{size}}\sigma^{OC}_{\text{size}}, \tag{10}$$

$$\sigma^{OC}_{\text{deg}} = w'^{OC}_{\text{lpips}}\sigma^{OC}_{\text{lpips,deg}} + w'^{OC}_{\text{dino}}\sigma^{OC}_{\text{dino,deg}} + w'^{OC}_{\ell 2}\sigma^{OC}_{\ell 2,\text{deg}}, \tag{11}$$

$$\sigma^{OC}_{\text{edge}} = w'^{OC}_{\text{lpips}}\sigma^{OC}_{\text{lpips,edge}} + w'^{OC}_{\text{dino}}\sigma^{OC}_{\text{dino,edge}} + w'^{OC}_{\ell 2}\sigma^{OC}_{\ell 2,\text{edge}}. \tag{12}$$

The weights with apostrophes (w'^{OC}_{*}) are optimized only for OBJECT ATTRIBUTE CHANGE, independently from the other weights from other tasks. The fifth point is measured with σ^{BC}. The original and edited backgrounds are obtained by masking the union of the original and the edited object region from I_0 and I_e. We mask out the same regions form I_0 and I_e for the same reason as in Appendix C.1.1. When any foreground object o is not detected from I_e, σ^{OF} and σ^{OC} is set to 0, and σ^{BC} is computed with backgrounds with only the original object regions masked. In sum, the OBJECT ATTRIBUTE CHANGE is evaluated with three scores, Object Fidelity σ^{OF}, Object Consistency σ^{OC}, and Background Consistency σ^{BC}.



(a) Size score

(b) Position score

Figure II. **Our (a) size scores and (b) a position score** plotted for relative size and position changes, respectively.

C.1.6 Background Change

We define three key points to satisfy for a good Background Change editing. 1) The background should be changed to the desired background (b), while 2) retaining all the foreground objects. The first point is quantified by σ^{BF}. To obtain the original and edited background, we mask out the union of all the foreground objects detected in I_0 and I_e, where we indicate the foreground objects as every editable object in the image, since those are what should be unchanged from C_0 to C_e.

The second point is measured by the average of $\sigma^{OC}_{o\in\mathcal{O}}$, where σ^{OC}_o is σ^{OC} in Eq. (3) of an object o in set of all foreground objects $\mathcal{O}$. σ^{OC} can be formulated as:

$$\sigma^{OC} = \frac{1}{|\mathcal{O}|}\sum_{o\in\mathcal{O}}(w^{OC}_{\text{lpips}}\sigma^{OC}_{\text{lpips},o} + w^{OC}_{\text{dino}}\sigma^{OC}_{\text{dino},o} + w^{OC}_{\ell 2}\sigma^{OC}_{\ell 2,o} + w^{OC}_{\text{pos}}\sigma^{OC}_{\text{pos},o} + w^{OC}_{\text{size}}\sigma^{OC}_{\text{size},o}). \tag{13}$$

When any foreground object o is not detected from I_e, σ^{OC}_o is set to 0, and σ^{BC} is computed without the mask of o. In sum, the Background Change task is evaluated by two scores, Background Fidelity σ^{BF} and Object Consistency σ^{OC}.

C.1.7 Style Change

Two main key-points that defines a good Style Change is whether the style of the image (I_0) is changed 1) to desired style (s), while 2) retaining fundamental morphological character of the image. The first point is quantified by σ^{BF}. Here, we use whole image I_0 and I_e for backgrounds. The second point is quantified by σ^{BC}_{deg} and σ^{BC}_{deg}, which are σ^{BC} computed with degraded and edged (explained in Appendix C.1.5) I_0 and I_e respectively. σ^{BC} for Style Change is formulated as:

$$\sigma^{BC} = w^{BC}_{\text{deg}}\sigma^{BC}_{\text{deg}} + w^{BC}_{\text{edge}}\sigma^{BC}_{\text{edge}}, \tag{14}$$

$$\sigma^{BC}_{\text{deg}} = w'^{BC}_{\text{lpips}}\sigma^{BC}_{\text{lpips,deg}} + w'^{BC}_{\text{dino}}\sigma^{OC}_{\text{dino,deg}} + w'^{BC}_{\ell 2}\sigma^{BC}_{\ell 2,\text{deg}}, \tag{15}$$

$$\sigma^{BC}_{\text{edge}} = w'^{BC}_{\text{lpips}}\sigma^{BC}_{\text{lpips,edge}} + w'^{BC}_{\text{dino}}\sigma^{BC}_{\text{dino,edge}} + w'^{BC}_{\ell 2}\sigma^{BC}_{\ell 2,\text{edge}}. \tag{16}$$

Scores with subscript "deg" and "edge" are corresponding scores computed with degraded and edged object segments respectively and weights with apostrophe are weights optimized to Style Change evaluation independently from the other weights from other task evaluations. In sum, Style Change edited image is evaluated by two scores, background fidelity σ^{BF} and background consistency σ^{BC}.

C.2. Evaluation Metrics

Fig. II describes our mathematic size score and position consistency score. For the size score, we first calculate the relative size change $(A_e/A_0)^{0.5}$, where A_0 is area of the object mask in the original image, and A_e is the area of the object mask in edited image. When we calculate the size score for edit fidelity, we give full score for changes greater than thresholds (r_1 and r_2 in Fig. IIa) that are empirically set to be the boundary where it shows noticeable size change. The size score is zero for changes opposite to the intended direction, and linearly scaled results are in between. When we compute the size score for consistency, we give the full score (1.0) for perfect size preservation. The size score is linearly scaled down to 0 as the relative

Correlation	Object Fidelity	Background Fidelity	Object Consistency	Background Consistency	Total Score
ρ	0.7000	0.6377	0.9710	1.0000	1.0000
τ	0.6000	0.5520	0.9309	1.0000	1.0000

Table I. **Correlation coefficients between model winning rates from user study and our metrics on the user study training set.** (See Tab. 2 for the result on the test set.)

size change changes from 1 to 0 and maximum possible change rate (r_3 in Fig. IIa), which is defined as $((H \times W)/A_0)^{0.5}$, where H and W indicate the height and width of the image, respectively. Position consistency is linearly scaled down from 1 to 0 as deviation of the object mask's center of mass relative to square root of original mask size increase from 0 to maximum possible value; that is, $= \sqrt{H^2 + W^2}/A_0^{0.5}$.

C.3. Fitting with Human Evaluation

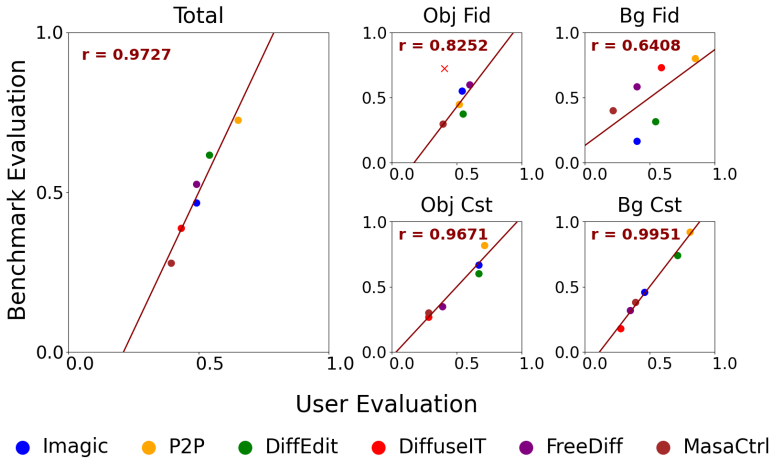


Figure III. **Relation between winning rates by users and HATIE, measured on the user study training set.** The least square linear fit (red line) and Pearson's correlation coefficient (r) are reported. (See Fig. 8 for the results on the test set.)

Fig. III plots the same figure as Fig. 8 but for the *training* results for our benchmark weight optimization. Each figure shows that our optimization process is close to ideal. However, the two fidelity scores are fitted slightly sub-optimally. This is because both object size fidelity and background fidelity are based on a single metric. This makes it impossible to attempt any weight optimization and restricts adjustability of our benchmark. This indirectly proves again the importance of diversity in scoring metrics to ensure robustness of the benchmark. The other two correlation coefficients, Spearman's ρ and Kendall's τ, are also presented in Tab. I.

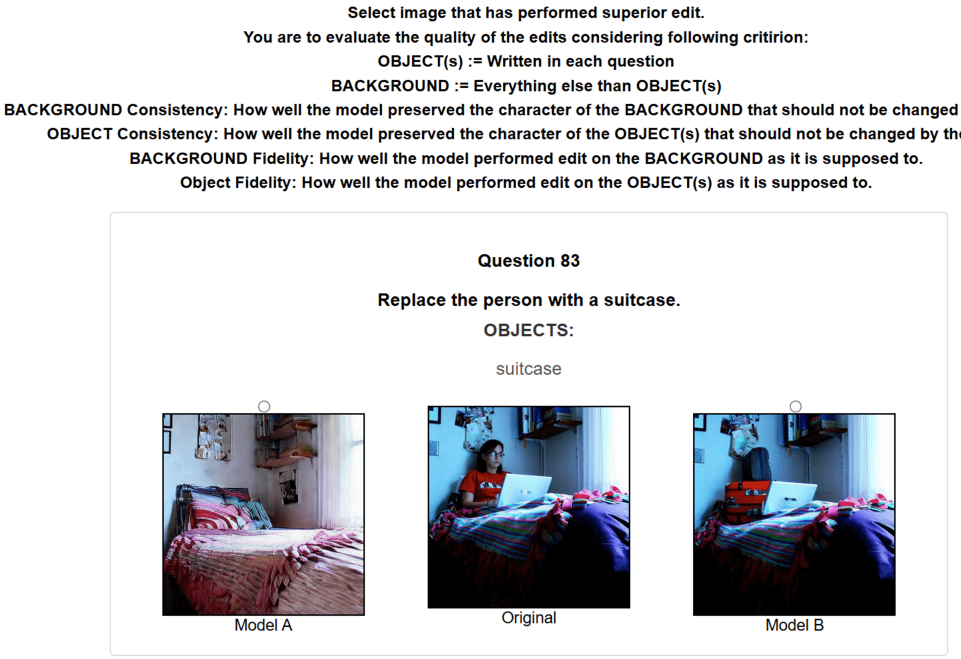
C.4. Examples of User Study Questions

Sample guidelines and questions of our user study are provided in Fig. IV. Participants were given with a brief instruction of the questions and evaluation criteria that they were supposed to take into account. Each question informs the participants what kind of edit has been performed on which object. Given all these information, participants were asked to choose a superior result out of two options provided with an original image.

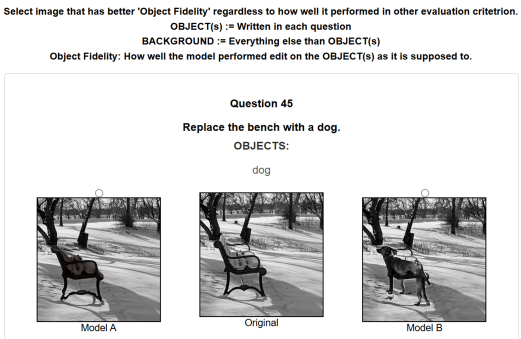
D. Additional Experimental Results

D.1. Suitability of Metric Models

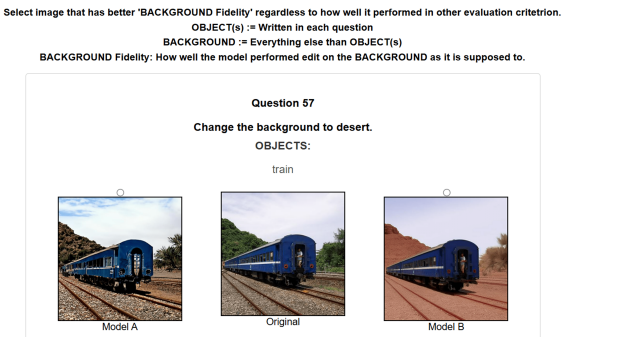
Fig. V shows samples of edited images by P2P across a range of τ, along with the trend of our evaluation metric scores. There are differences in the sensitivity and amplitude of changes, but overall, we observe that all metrics agree with our design intention. As edits are getting more faithful, all the metric scores involved in fidelity are increasing. Also, as the edit



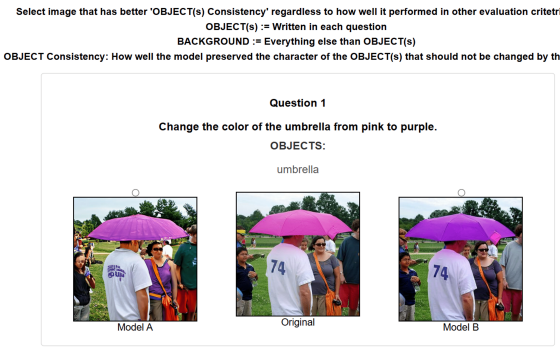
(a) Total Score (Overall Edit Quality)



(b) Object Fidelity



(c) Background Fidelity



(d) Object Consistency

(e) Background Consistency

Figure IV. **User study question examples.** Each question is binary, asking to choose the better edited result for each score criterion. The target object in each image is also given for the user to help evaluate. (a) is one of overall edit quality (corresponding to the TOTAL SCORE) assessment question, selecting the better edited image considering all evaluation criteria. (b) is a question for Object Fidelity (OF), which is focused on selecting an image that has better quality of the editable object. (c) is for measuring Background Fidelity (BF), selecting an image with better quality of background along instruction. (d) is for Object Consistency (OC), selecting an image that preserves the original object better. (e) is for Background Consistency (BC), which is focused on selecting image that preserves the original background better.

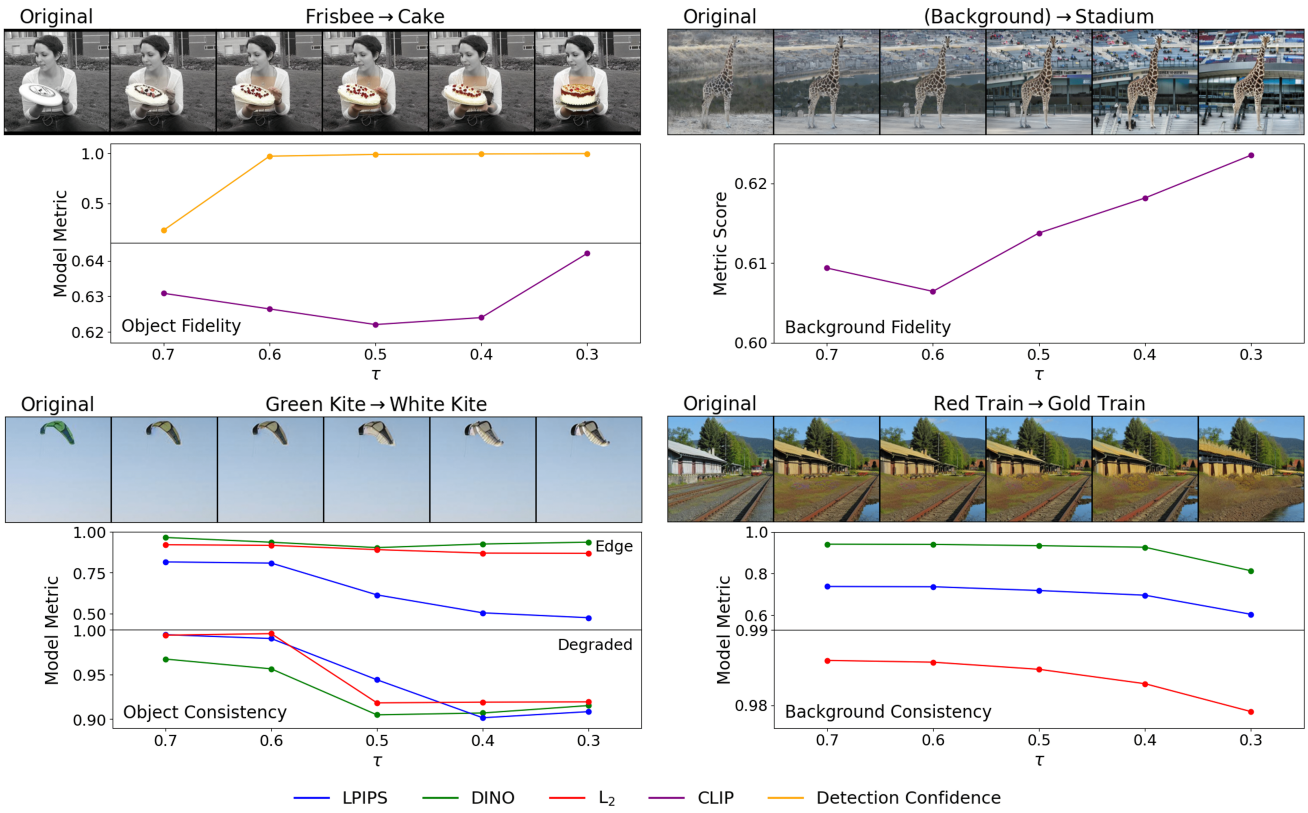


Figure V. **Evaluations by each metric for edited images with different edit strength.** The editing has been done using the P2P model with various hyper-parameter τ, where a higher τ means weaker editing, and the paired graph illustrates the specific metric scores of our method. In each example, the left-most image is the original image and the rest shows the edited images for the query indicated above, using different τ values indicated below. On the top-left, the Object Fidelity (OF) scores are plotted, and the top-right quadrant shows the Background Fidelity (BF) metrics. At the bottom-left, the Object Consistency (OC) scores measured for a OBJECT ATTRIBUTE CHANGE task is shown. The two figures are computed from detected Canny-edge of the objects and degraded object images, respectively. Lastly, the bottom-right graph shows Background Consistencies (BC) measured with different metric models.

result gradually fails to retain the original character, the two consistency scores in the second row clearly drop. These results show that our metrics capture the characteristics and quality of the edited images well, and therefore, are suitable for editing benchmark.

D.2. Reliability of Instance Segmentation Model

The credibility of our evaluation method is strongly dependent on that of the instance segmentation model we use. To avoid detection failures, during the evaluation process, we gather all the detection results, even those with extremely low confidence. However, this does not guarantee a perfect detection. The overall failure ratio of the instance segmentation model, where it could not find the desired object in the image, was 16.01%. Also, in many cases, these failures were not because of the limitation of the segmentation model. We demonstrate 30 randomly sampled images in Fig. VI with the object names that the detector had tried to locate. From this figure, we see that the detection failure is usually due to the absence of desired object, not because of the model's limitation. Thus, we claim that the instance segmentation used in our experiment is sufficiently credible to conduct the evaluation.

D.3. Detailed Results

Fig. VII plots a similar figure to Fig. 7 but across various models, where we see that our HATIE gives accurate assessments aligned with real images. Both fidelity scores are low for failure cases, while they get higher on successful ones. Consistency

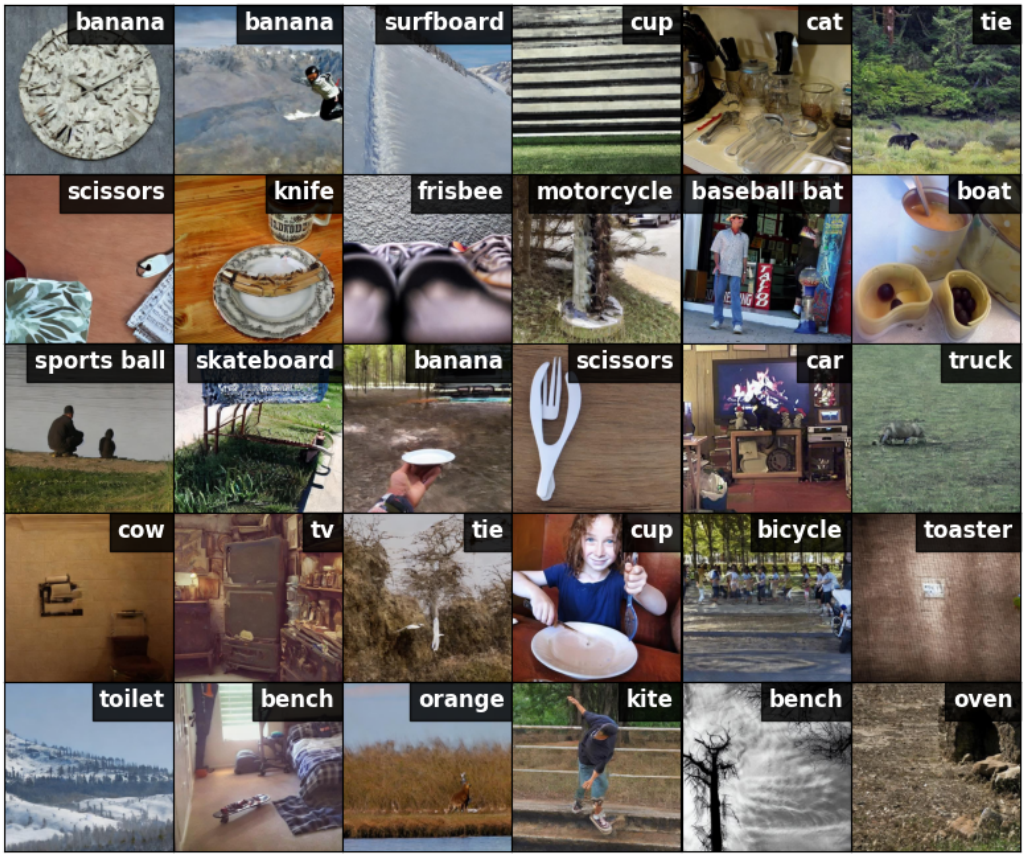


Figure VI. **Illustration of segmentation failure cases.** Images of instance segmentation failures are plotted together with the object names the detector tried to locate.

scores for those that excellently retains the original details, that partially retains the original details, and that completely loses the original details are clearly distinguished throughout the model scores. Total score shows a balance between these criteria, where the highest score is given to the image that not only changes the object correctly, but also keeps the background perfectly.

Fig. VIII plots the same graph with Fig. 6 but for result of one of the instruction-based model, InstructPix2Pix. We can also observe identical trends found in Fig. 6. Fig. IX plots the same data listed in Tab. 5 and 6. Here we can visually see strong points and weak points of each model together with inter-model comparison. Tab. II gives full list of every model benchmark scores on every tested parameter settings for all 5 subscores and total scores.

D.4. Experiment on human perception alignment with additional dataset

The alignment results of the unseen dataset in Figure X is presented in the same format as Figure 8 in the main paper. We tested our metric and human preference alignment on out-of-domain dataset, ImageNet[7]. This result demonstrates that HATIE maintains high correlation with human evaluations even on a unseen dataset. This suggests that HATIE can be extended to various types of data and applied to a wide range of editing tasks in the future.

E. Limitations

Our benchmark aims to encompass every feasible editing task, but some tasks have been inevitably excluded. First, OBJECT REMOVAL has been limited only for the instruction-based models, due to the difficulty generating captions for description-based models, as explained in Sec. 2. Specifically, 8,315 OBJECT REMOVAL queries are carried out with the three instruction-based models. The evaluation workflow of OBJECT REMOVAL task for the instruction-based models is shown in Fig. XI, and the benchmark results are listed in Tab. III.

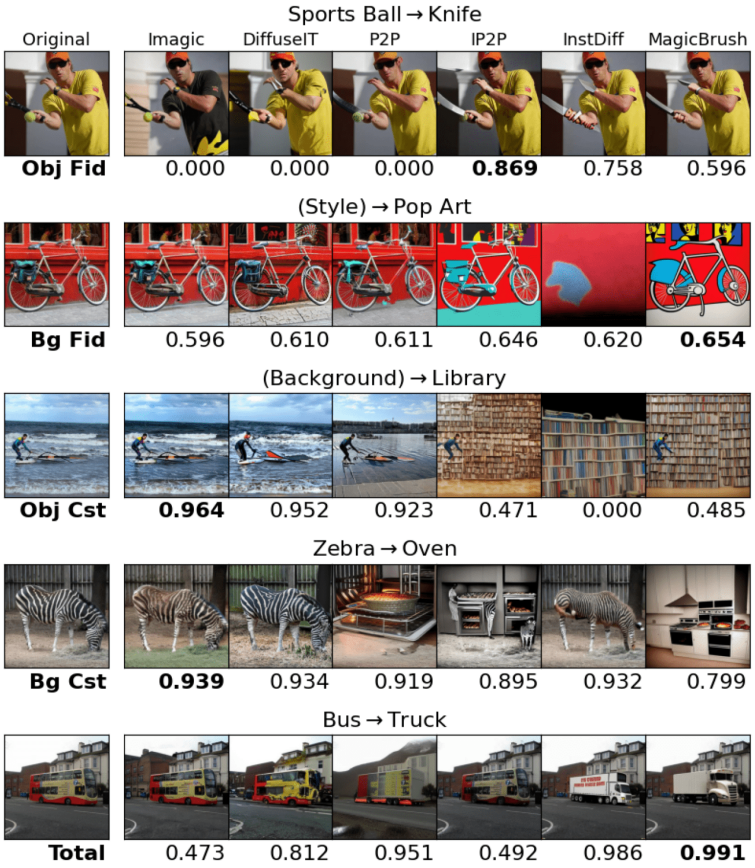


Figure VII. **Examples of our evaluation results.** The left-most image is the original input image. The next three shows the output image and evaluated metrics for select description-based models, while the last three are those for instruction-based models, according to the query indicated above each example.

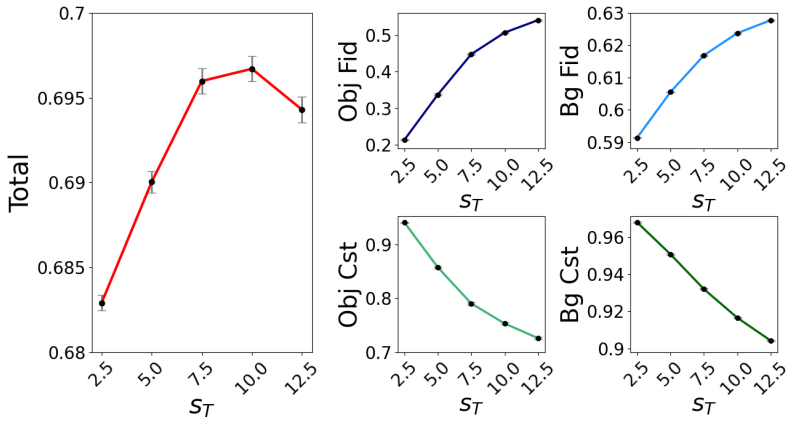
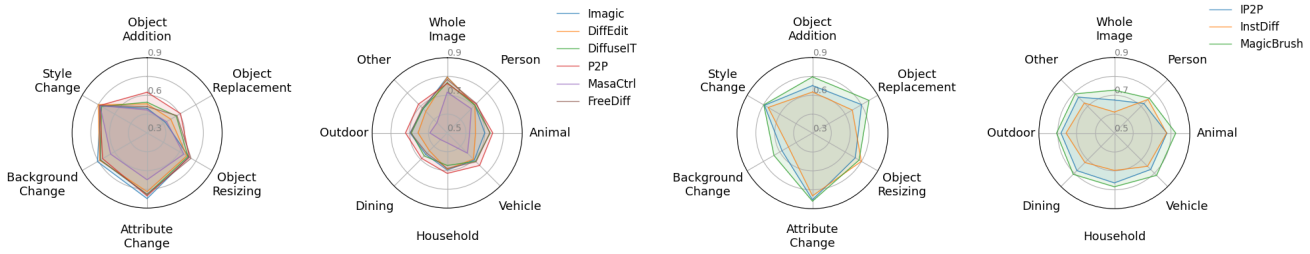


Figure VIII. **Benchmark scores with varied editing intensity for Instruct-Pix2Pix.** Each figure plots each metric and the Total Score for a range of hyper-parameter values.

Second, moving or rotating an object has not been considered. In order to clearly define where or which direction to move or rotate an object, not only the target object to move or rotate but also another object to become a reference point of a new location or direction are needed. However, our base dataset contain few images containing two or more editable objects,



(a) Description-based Models

(b) Instruction-based Models

Figure IX. **Evaluation scores for each edit type and edit target object class for each model.** (a) plots for the six description-based models and (b) plots for the three instruction-based models.

	Models		Object Fidelity	Background Fidelity	Object Consistency	Background Consistency	Image Quality	Total Score
Description -based	DiffEdit		0.2277 ± 0.0018	0.5910 ± 0.0001	0.8338 ± 0.0018	**0.9608 ± 0.0002**	0.7477 ± 0.0024	0.6552 ± 0.0006
	DiffuseIT		0.3202 ± 0.0019	0.6045 ± 0.0001	0.8616 ± 0.0012	0.8958 ± 0.0002	0.5569 ± 0.0021	0.6682 ± 0.0006
	FreeDiff	$s_T = 2.5$	0.2571 ± 0.0018	0.5903 ± 0.0001	0.8980 ± 0.0011	0.9413 ± 0.0002	0.8360 ± 0.0014	0.6749 ± 0.0005
		$s_T = 5.0$	0.2954 ± 0.0020	0.5939 ± 0.0001	0.8766 ± 0.0012	0.9253 ± 0.0002	0.7770 ± 0.0016	0.6749 ± 0.0006
		$s_T = 7.5$	0.3174 ± 0.0020	0.5964 ± 0.0001	0.8631 ± 0.0013	0.9150 ± 0.0002	0.7180 ± 0.0014	0.6739 ± 0.0006
		$s_T = 10.0$	0.3323 ± 0.0020	0.5982 ± 0.0001	0.8545 ± 0.0013	0.9074 ± 0.0002	0.6651 ± 0.0020	0.6729 ± 0.0006
		$s_T = 12.5$	0.3405 ± 0.0021	0.5995 ± 0.0001	0.8487 ± 0.0013	0.9014 ± 0.0002	0.6185 ± 0.0028	0.6714 ± 0.0006
	P2P	$\tau = 0.3$	**0.3595 ± 0.0020**	**0.6073 ± 0.0002**	0.8528 ± 0.0015	0.9261 ± 0.0003	0.6551 ± 0.0023	0.6858 ± 0.0006
		$\tau = 0.4$	0.3334 ± 0.0019	0.6038 ± 0.0002	0.8705 ± 0.0014	0.9339 ± 0.0003	0.7097 ± 0.0031	**0.6859 ± 0.0006**
		$\tau = 0.5$	0.3063 ± 0.0019	0.6002 ± 0.0001	0.8811 ± 0.0013	0.9397 ± 0.0003	0.7551 ± 0.0029	0.6833 ± 0.0006
		$\tau = 0.6$	0.2823 ± 0.0018	0.5971 ± 0.0001	0.8864 ± 0.0013	0.9433 ± 0.0003	0.7830 ± 0.0027	0.6794 ± 0.0006
		$\tau = 0.7$	0.2630 ± 0.0018	0.5946 ± 0.0001	0.8898 ± 0.0013	0.9456 ± 0.0003	0.7993 ± 0.0023	0.6758 ± 0.0005
	Imagic	$\eta = 0.2$	0.2138 ± 0.0016	0.5866 ± 0.0001	**0.9380 ± 0.0008**	0.9421 ± 0.0003	0.8499 ± 0.0018	0.6737 ± 0.0004
		$\eta = 0.4$	0.2179 ± 0.0017	0.5869 ± 0.0001	0.9311 ± 0.0009	0.9339 ± 0.0003	**0.8519 ± 0.0019**	0.6711 ± 0.0005
		$\eta = 0.6$	0.2271 ± 0.0017	0.5878 ± 0.0001	0.9237 ± 0.0009	0.9201 ± 0.0003	0.8406 ± 0.0014	0.6682 ± 0.0005
		$\eta = 0.8$	0.2534 ± 0.0018	0.5902 ± 0.0001	0.9134 ± 0.0010	0.9038 ± 0.0004	0.8184 ± 0.0020	0.6682 ± 0.0005
		$\eta = 1.0$	0.2927 ± 0.0019	0.5970 ± 0.0001	0.8942 ± 0.0012	0.8841 ± 0.0003	0.7678 ± 0.0034	0.6690 ± 0.0006
	MasaCtrl	$s_T = 2.5$	0.1833 ± 0.0017	0.5932 ± 0.0001	0.6312 ± 0.0026	0.9042 ± 0.0003	0.2580 ± 0.0032	0.5716 ± 0.0008
		$s_T = 5.0$	0.2111 ± 0.0018	0.5949 ± 0.0001	0.6586 ± 0.0024	0.9003 ± 0.0003	0.2598 ± 0.0030	0.5846 ± 0.0007
		$s_T = 7.5$	0.2332 ± 0.0019	0.5967 ± 0.0001	0.6754 ± 0.0023	0.8954 ± 0.0003	0.2685 ± 0.0022	0.5936 ± 0.0007
		$s_T = 10.0$	0.2496 ± 0.0019	0.5983 ± 0.0001	0.6868 ± 0.0023	0.8908 ± 0.0003	0.2834 ± 0.0029	0.5999 ± 0.0007
		$s_T = 12.5$	0.2632 ± 0.0020	0.5995 ± 0.0001	0.6982 ± 0.0022	0.8867 ± 0.0002	0.2968 ± 0.0025	0.6056 ± 0.0007
Instruction -based	MagicBrush		0.5378 ± 0.0020	0.6196 ± 0.0002	0.8259 ± 0.0018	0.9513 ± 0.0003	0.6977 ± 0.0032	**0.7329 ± 0.0007**
	InstDiff		0.4596 ± 0.0022	0.6205 ± 0.0001	0.6870 ± 0.0022	0.9090 ± 0.0005	0.4148 ± 0.0039	0.6639 ± 0.0008
	IP2P	$s_T = 2.5$	0.2141 ± 0.0016	0.5914 ± 0.0001	**0.9407 ± 0.0008**	**0.9678 ± 0.0002**	**0.8983 ± 0.0014**	0.6829 ± 0.0005
		$s_T = 5.0$	0.3373 ± 0.0019	0.6055 ± 0.0002	0.8573 ± 0.0017	0.9506 ± 0.0003	0.8049 ± 0.0024	0.6900 ± 0.0006
		$s_T = 7.5$	0.4474 ± 0.0021	0.6169 ± 0.0002	0.7903 ± 0.0021	0.9319 ± 0.0003	0.6658 ± 0.0031	0.6960 ± 0.0007
		$s_T = 10.0$	0.5064 ± 0.0020	0.6237 ± 0.0001	0.7531 ± 0.0023	0.9164 ± 0.0004	0.5394 ± 0.0035	0.6967 ± 0.0008
		$s_T = 12.5$	0.5403 ± 0.0020	0.6277 ± 0.0001	0.7258 ± 0.0023	0.9041 ± 0.0004	0.4393 ± 0.0034	0.6943 ± 0.0008
		$s_I = 1.0$	**0.6002 ± 0.0020**	**0.6342 ± 0.0001**	0.6640 ± 0.0025	0.8762 ± 0.0004	0.2839 ± 0.0026	0.6855 ± 0.0008
		$s_I = 1.25$	0.5385 ± 0.0020	0.6273 ± 0.0001	0.7213 ± 0.0023	0.9092 ± 0.0004	0.4858 ± 0.0029	0.6948 ± 0.0008
		$s_I = 1.5$	0.4474 ± 0.0021	0.6169 ± 0.0002	0.7903 ± 0.0021	0.9319 ± 0.0003	0.6658 ± 0.0031	0.6960 ± 0.0007
		$s_I = 1.75$	0.3547 ± 0.0020	0.6060 ± 0.0002	0.8521 ± 0.0018	0.9467 ± 0.0003	0.7831 ± 0.0032	0.6918 ± 0.0007
		$s_I = 2.0$	0.2862 ± 0.0018	0.5983 ± 0.0001	0.8943 ± 0.0014	0.9547 ± 0.0003	0.8367 ± 0.0021	0.6865 ± 0.0006

Table II. **HATIE results across all score criteria.** Our HATIE evaluation results are shown for the six description-based models and three instruction-based models, across five score criteria. We also show the differences between various hyper-parameter values in each model.

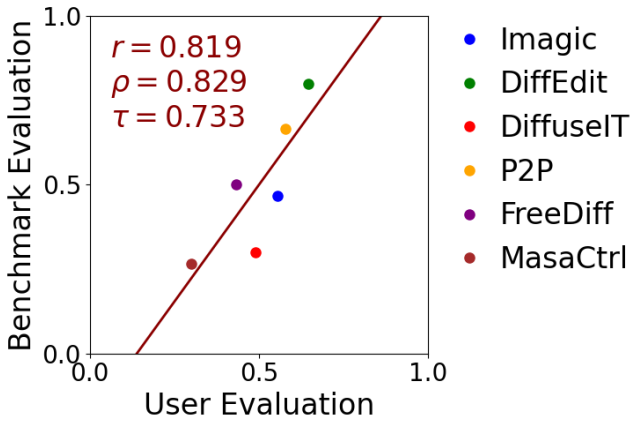
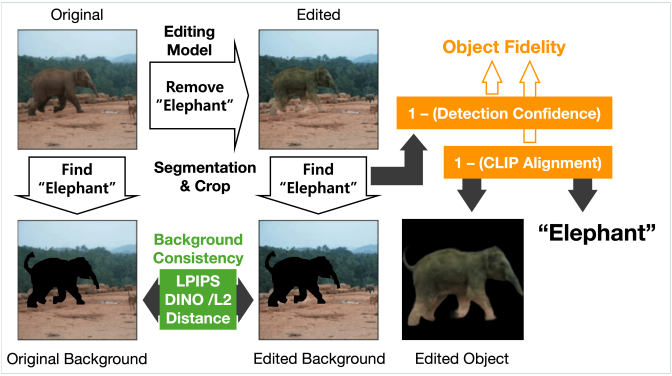
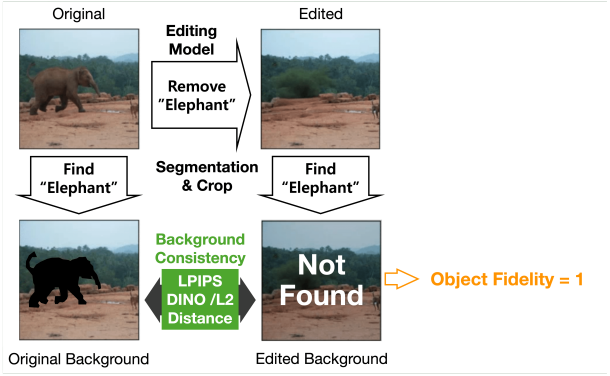


Figure X. Alignment between human preference and HATIE on unseen new dataset.



(a) Target object found



(b) Target object not found

Figure XI. **Evaluation workflow of OBJECT REMOVAL query result.** (a) shows the case when target object is found in edited image, and (b) shows the case when target object is not found in edited image.

making it challenging to create a large number of movement or rotation queries. We leave it as a future work which should be addressed with an additional base dataset.

Lastly, for the OBJECT ATTRIBUTE CHANGE queries, GQA annotations have many other attribute words beyond our four classes (color, state, material, action), such as attributes about emotion, hair, or fashion. We decide to exclude these attributes mainly because of the ambiguity to localize the exact area to be affected by changing these attributes, resulting in difficulty to measure fidelity and consistency. To overcome this, we may divide the human figure into more manageable regions (such as the face, hair, body, arms, and legs) using human pose estimation. This strategy may widen the scope of our benchmark, but we leave it as a potential future work.

Models		Object Removal
MagicBrush		0.7230 ± 0.0025
InstructDiffusion		$\mathbf{0.9082 \pm 0.0016}$
Instruct-Pix2Pix	$s_T = 2.5$	0.5272 ± 0.0010
	$s_T = 5.0$	0.5834 ± 0.0019
	$s_T = 7.5$	0.6485 ± 0.0024
	$s_T = 10.0$	0.6817 ± 0.0025
	$s_T = 12.5$	0.6925 ± 0.0025
	$s_I = 1.0$	0.6972 ± 0.0025
	$s_I = 1.25$	0.7159 ± 0.0026
	$s_I = 1.5$	0.6485 ± 0.0024
	$s_I = 1.75$	0.5694 ± 0.0018
	$s_I = 2.0$	0.5369 ± 0.0013

Table III. **Result of OBJECT REMOVAL task** in instruction-based models on HATIE benchmark.