This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Towards Scalable Human-aligned Benchmark for Text-guided Image Editing

Suho Ryu*, Kihyun Kim*, Eugene Baek*, Dongsoo Shin, Joonseok Lee†
Graduate School of Data Science, Seoul National University
{jmhera2007,ki5477,eugene0103,dongsoo,joonseok}@snu.ac.kr

Abstract

A variety of text-guided image editing models have been proposed recently. However, there is no widely-accepted standard evaluation method mainly due to the subjective nature of the task, letting researchers rely on manual user study. To address this, we introduce a novel Human-Aligned benchmark for Text-guided Image Editing (HATIE). Providing a large-scale benchmark set covering a wide range of editing tasks, it allows reliable evaluation, not limited to specific easy-to-evaluate cases. Also, HATIE provides a fully-automated and omnidirectional evaluation pipeline. Particularly, we combine multiple scores measuring various aspects of editing so as to align with human perception. We empirically verify that the evaluation of HATIE is indeed human-aligned in various aspects, and provide benchmark results on several state-of-the-art models to provide deeper insights on their performance.

1. Introduction

With recent advances in diffusion-based image generation models [25–27], text-guided image editing has been prominently developed. For example, Imagic [15], Prompt-to-Prompt [10], and SDEdit [20] have pioneered this direction in early years, and recent models [9, 13, 34] have remarkably improved the performance.

However, there is no widely-accepted standard evaluation methodology to accurately and objectively assess these models. This partly comes from the nature of the editing task that there can be multiple correct outputs for each instance. For example, "adding a table in a living room scene" can be satisfied with many different shapes of the table. Even for a more specific query like "a circular wooden tea table with three legs", the pixel-level details can be infinitely diverse. This becomes even worse when the query is more ambiguous.

This inherent nature of text-driven image editing causes difficulties in evaluation. Due to the lack of a 'golden' ground truth, it is crucial to have a proper metric to measure the deviation between the edited image and the ground truth. Unlike traditional machine learning problems where we can easily measure the deviation using a simple distance metric like L_2, it is well-known that image distance cannot be simply measured by pixel-level distances. Thus, it is essential to have *human-like high-level understanding* to measure the relevance with human perception to properly evaluate the quality of edited images. For this, model-based metrics such as CLIP [24] similarity or LPIPS [35] distance has been used. Also, large language models (LLM) or vision language models (VLM) have been widely adopted recently. Relying on a particular model, however, sacrifices reliability and reproducibility of evaluation results.

In addition, there are *multiple aspects* to consider when we evaluate the edited images. First of all, the edited image itself should be realistic and feasible. The overall configuration and specific details of each object in the image should be consistent with our common knowledge. Second, the output image should correctly reflect the instructed change. At the same time, it is expected not to alter unintended aspects in the instruction. For instance, if an image shows a person and the task is to make her smile, the identity of the person should not change after editing (Fig. 1). Due to these multi-faceted aspects of image editing, a crude evaluation would be insufficient to assess detailed quality of the edit. As it is unclear how to automate this overall evaluation, most previous research have relied on user study, which is fundamentally limited to scale and reproduce.

There have been a few notable attempts to automate evaluation of image editing, listed in Tab. 1. Kawar et al. [15] and Wang et al. [30] presented a benchmark guideline and an image set. Due to their small *scale of the benchmark set*, however, this work is insufficient to perform robust evaluation, and thus they also have relied on user study. Shi et al. [28] provided a set of editing requests with ground truth result image for automated evaluation. However, their editing tasks are quite limited to those where a clear ground truth can be obtained at almost pixel-level, *e.g.*, gray scaling an image. Aiming at a similar goal to ours, Basu et al. [2] diversified editing tasks and target objects to cover a wider

*Equal Contribution
†Corresponding author



(a) Original image (b) Edited images to "make her smile"

Figure 1. **An example highlighting the importance of consistency in image editing.** (a) Original image (b) Edited images for a prompt "Make her smile". The left result is more consistent with the input than the right one, better preserving her identity.

Method	# Images	# Queries	# Obj. class	Auto-eval
TEdBench [15]	100	100	N/A	No
EditBench [30]	240	720	N/A	N/A
GIER [28]	6,179	30,895	N/A	N/A
EditVal [2]	92	648	19	Partial
HATIE (Ours)	**18,226**	**49,840**	**76**	**Full**

Table 1. **Comparison with other image editing benchmarks.** HATIE provide a fully-automated large-scale evaluation method covering wide range of objects and tasks.

range of editing. The evaluation, however, has been done by user study.

In this paper, we propose a scalable **H**uman-**A**ligned **T**ext-guided **I**mage **E**diting benchmark (**HATIE**) that addresses the aforementioned limitations. First, our large-scale benchmark set *covers a wide range of editing cases*, not limited to specific easy-to-evaluate cases. Second, HATIE measures the quality of edited images in a fully-automated and *multifaceted* way, *aligned with human perception*. Our benchmark allows robust, objective, and labor-free evaluation of existing and new image editing models, paving the road for the advancement of this field.

2. Problem Formulation

Given an image I_0 and a natural language caption C, the text-guided image editing task aims to generate an edited image I_e, which faithfully reflects the instruction described in C while minimizing unintended deviations from I_0. A text-guided image editing mode f may take an additional parameter p to adjust the degree of manipulation, focusing more (or less) on reflecting the instruction C compared to preserving the context. Formally, $I_e = f(I_0, C; p)$.

Depending on the type of the caption C, text-guided image editing models can be divided into two categories: description-based and instruction-based. Description-based models are conditioned by two separate captions, $C = \{C_0, C_e\}$, where C_0 and C_e describe the original and desired images, respectively. For example, given C_0 = "a photo of a dog" and C_e = "a photo of a cat", the editing task is to replace the dog in I_0 with a cat. Many early approaches [1, 5, 6, 10, 15, 16, 19, 22, 31] have adopted this scheme, as it is straightforward to directly apply an image generation model conditioned on a text caption. However, this approach has drawbacks. Certain types of editing queries cannot be easily articulated with two separate captions. For example, to remove a dog from a photo, C_0 can be "A photo of a dog" but C_e would be "A photo without a dog", which is not a clear description of the desired image.

On the other hand, instruction-based models [3, 9, 29, 34] take a single instruction C, describing the desired delta between the original and the edited image. Under this setting, with an instruction C = "remove the dog", a model is expected to generate an image I_e without the dog from I_0.

Our framework HATIE is applicable to both caption types. Note that models taking different types of captions are not comparable. It might be possible to convert one type of captions to the other, but evaluating on this indirect setting would be unfair for one side.

3. The HATIE Framework

For a fair comparison of editing models, a same set of queries should be given to every model, and the set should be large enough to cover a wide range of possible domains, so that no model is particularly favored. Also, the results should be evaluated by consistent, objective, and equitable metrics. Moreover, users should be able to consistently reproduce the evaluation results at any time. For this, it is discouraged to rely on external services (*e.g.*, ChatGPT) without public access to the actual parameters for evaluation.

In order to satisfy these conditions, our HATIE provides a large-scale comprehensive dataset, containing images to be edited and annotations of editable objects in them (Sec. 3.1), a set of curated editing queries for these editable objects, in both description and instruction forms (Sec. 3.2), and a set of evaluation metrics designed to evaluate the edited images from various perspectives (Sec. 3.3). Fig. 2 illustrates a diagram of the overall benchmark workflow.

3.1. Dataset

Base Dataset. We take the GQA dataset [14], one of foundational datasets designed for Visual Question and Answering (VQA) tasks as our benchmark base. It consists of 113K real-world images with in-depth annotations, including the list of objects labeled with their names, bounding boxes, attributes, and relations with other objects (scene graph). As a precise visual understanding is crucial to figure out what kind of editing jobs are feasible in each image, rich annotations in GQA make it suitable for generating edit queries.

Data Filtering and Augmentation. Not every object in a scene is suitable for editing. We filter out indistinguishable objects that are too small, cropped, or occluded. Also, for automated evaluation, it is necessary for an object to

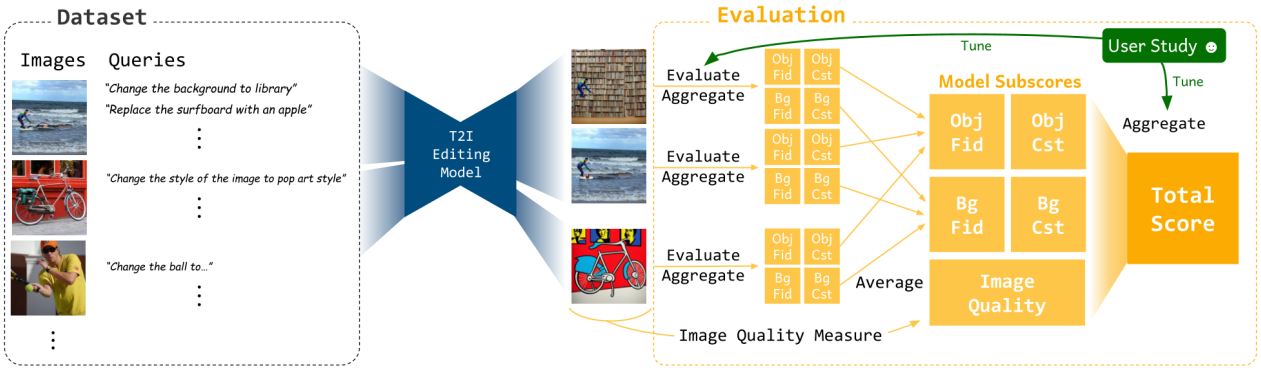


Figure 2. **Overview of our HATIE Benchmark.** HATIE consists of an image and query dataset for editing, along with an automated evaluation pipeline for assessing editing performance. We curate a large-scale comprehensive dataset with images and corresponding editing queries, on which a model would perform text-guided image editing. Then, HATIE evaluates the edited images from 5 different aspects: *Object Fidelity, Background Fidelity, Object Consistency, Background Consistency,* and *Image Quality*. Finally, these scores are aggregated by a weight fitted to human feedback through our user study, producing the final *Total Score.*

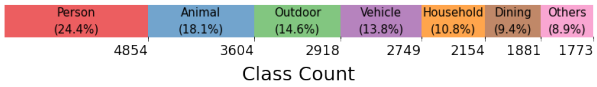


Figure 3. **Object Class Distribution in Our Dataset.** HATIE evaluates fairly by providing evenly distributed dataset.

be detected by object detector, which is typically trained on a dataset labeled for a fixed set of classes, *e.g.*, MS-COCO [18]. We first match the freeform GQA object names with 80 object classes in COCO, and filter out objects not included in the COCO classes. Then, we remove the objects that are not detected by an instance segmentation model [32] used at evaluation (Sec. 3.3). We also exclude images containing two or more instances belonging to the same class to avoid any ambiguity in figuring out which one is the target to be edited. More details on dataset filtering is in Appendix A.

Then, the remaining images are trimmed and resized to 512×512 to be compatible with editing models. Since both editable objects and background are important for editing, we preserve the original image as much as possible, and make sure that editable objects are not cropped.

In total, our benchmark set consists of 18,226 images with 19,933 editable objects within 76 COCO classes. More detailed object class distribution is plotted in Fig. 3 and Appendix B.4.

3.2. Editing Query Formulation

Query Templates. On the collected images and objects, we generate queries to edit them. To automate this process but still make the queries realistic, we categorize the editing tasks into object-centric and non-object-centric.

An object-centric editing query focuses on a specific object, such as adding an object that is not present in the image (OBJECT ADDITION), removing an existing object (OBJECT REMOVAL), replacing an object with something else (OBJECT REPLACEMENT), and altering some attributes (OBJECT ATTRIBUTE CHANGE) or size (OBJECT RESIZING) of a particular object. More formally, given a set $\mathcal{C}$ of object classes and an image with editable object o in class $c_o \in \mathcal{C}$ with attributes $\{a_i\} \in \mathcal{A}$, we define templates for these object-centric edits as follows:

1. OBJECT ADDITION: Create an additional object o' in class $c_{o'} \in \mathcal{C}$ located at $r_{o'} \in \mathcal{R}$ relative to o.
2. OBJECT REMOVAL: Remove the object o from the image, generating a natural scene just without o.
3. OBJECT REPLACEMENT: Replace o with another object o' in class $c_{o'} \in \mathcal{C}$.
4. OBJECT ATTRIBUTE CHANGE: Change the attribute a_i of o to $a_j \in \mathcal{A}$.
5. OBJECT RESIZING: Make o larger (or smaller).

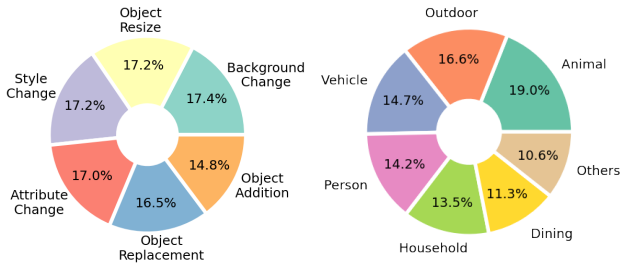
To construct the options, *i.e.*, $\mathcal{R}$ and $\mathcal{A}$, we employ a data-driven approach. We gather every attribute of each object class $c \in \mathcal{C}$ and every relation between each class pair $(c_1, c_2) \in \mathcal{C} \times \mathcal{C}$ from the annotations in the entire dataset. We classify and bind the collected attributes into 4 categories (color, state, material, and action), constructing a set of corresponding attributes: $\mathcal{A}_{\text{color}}$, $\mathcal{A}_{\text{state}}$, $\mathcal{A}_{\text{mat}}$, and $\mathcal{A}_{\text{act}}$. Also, we consolidate the freeform relation captions into 9 relations: $\mathcal{R} = \{$under, above, in, on, left, right, next to, front, behind$\}$. These attributes and relations cover most cases observed in the dataset.

On the other hand, a non-object-centric query involves background or style changes affecting the entire image:

1. BACKGROUND CHANGE: Change the background of the image to $b \in \mathcal{B}$.
2. STYLE CHANGE: Change the image style to $s \in \mathcal{S}$.

$\mathcal{B}$ and $\mathcal{S}$ are the set of options for background and style changes. Using a similar approach like above, we collect 27 distinct backgrounds in $\mathcal{B}$ and 10 styles in $\mathcal{S}$.

Feasibility of Queries. A query should be carefully designed, since feasibility of a specific edit depends on many factors in image such as position and size of the objects, relations between the objects or even view point and angle of the image. For instance, adding a surfboard *inside* a per-



(a) Edit Type Distribution (b) Target Class Distribution

Figure 4. **Query Set Distribution.** (a) Distribution of edit types in our query set, (b) Distribution of the object classes designated as the target in object-centric queries.

son would not make sense. Adding the surfboard *under* the person sounds reasonable, but it is impossible if it is out of the view. While manually defining queries for each image would yield the most accurate and diverse set, it is highly labor-intensive and may introduce subjective bias.

In order to restrict the queries within feasible options, we conduct a statistical analysis on the annotated relations between objects. We identify other relevant object types and relative locations that are frequently observed for each class, and restrict the options for the target object o' to this commonly observed ones. In OBJECT ADDITION, for example, we may add a *laptop* on the table (a frequently observed relation), but not a *car* (a rarely observed relation). As another example with OBJECT ATTRIBUTE CHANGE, the original attribute (*e.g.*, a_i = 'brown' $\in \mathcal{A}_{\text{color}}$) is replaced with another one (*e.g.*, a_j = 'yellow') from the same category ($\mathcal{A}_{\text{color}}$). We detail all task-specific considerations to make the queries feasible in Appendix B.1.

For non-object-centric tasks, on the other hand, we have relatively less restrictions. Changing the background to a random place may look odd but it is hardly impossible to imagine. Thus, we randomly select a background or style from pre-defined candidate sets of 27 backgrounds in $\mathcal{B}$ and 10 styles in $\mathcal{S}$. The full list of candidate backgrounds and styles are in Appendix B.1.

Considering these feasibility restrictions, we have generated 49,840 editing queries on the 18,226 images in total, balanced across various query types and options (Fig. 4, Appendix B.4).

Description and Instruction Generation. Finally, we formulate the natural language queries into an appropriate form, either description-based or instruction-based, using LLMs [8, 33]. See Appendix B.2 for the detailed generation flow and prompts.

3.3. Evaluation Criteria and Metrics

A good edit means that the target of the edit is faithfully modified according to the user's intention, while non-target elements remain unaltered. Therefore, we design the evaluation based on three criteria. First, we assess the general quality of the output images (**Image Quality**), as an image with poor quality itself is an indicator of suboptimal generation. Second, it is important to faithfully modify the image according to the instruction (**Fidelity**). Lastly, it is also important to verify that the model preserves the context of the original image except for the intended change (**Consistency**). With the help of an instance segmentation model [32], we further distinguish the latter two for the object and background. In total, we evaluate in 5 criteria: *Image Quality, Object Fidelity, Background Fidelity, Object Consistency,* and *Background Consistency*.

Image Quality. We measure the overall Image Quality (IQ) σ^{IQ} through Fréchet Inception Distance (FID) [12]. FID is calculated between input image set and corresponding output image set from the model, and rescale it to range $[0, 1]$ by

$$\sigma^{IQ} = 1 - \tanh(\text{FID}/25). \tag{1}$$

The scaling factor 25 is to make the score close to 1 for the state-of-the-art models.

Fidelity. For Object Fidelity (OF), we combine the following three metrics. First, we compute CLIP alignment [11] (CLIP embedding similarity between text and image) $\sigma^{OF}_{\text{clip}}$ between the target object and the desired target text. Specifically, we detect the target object o inside the edited image I_e using an instance segmentation model [32], and zero out the rest (later called the background). Then, we cut out the target object region with a minimal square enclosing it, denoted by M_o. We compute the CLIP alignment $\sigma^{OF}_{\text{clip}}$ between M_o and o. For example, if the query is to recolor a hat to red, we detect and separate the hat (M_o) from the output image I_e and calculate CLIP similarity to the target texts (o) such as 'hat' and 'red hat'.

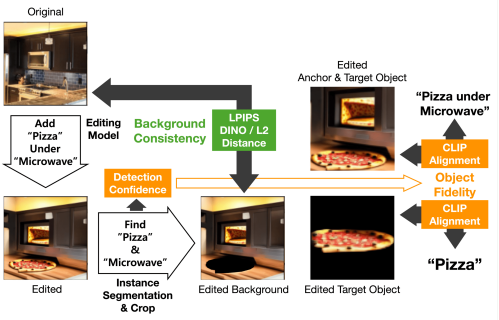
Second, we take the detection confidence σ^{OF}_{det} from the instance segmentation model for target object, *i.e.*, the hat. Lastly, we compute size fidelity $\sigma^{OF}_{\text{size}}$ for the OBJECT RESIZING queries, quantifying how much size of the object is appropriately adjusted through a rule-based formula described in Appendix C.2. The overall Object Fidelity score σ^{OC} is a convex combination of the three metrics:

$$\sigma^{OF} = w^{OF}_{\text{clip}}\sigma^{OF}_{\text{clip}} + w^{OF}_{\text{det}}\sigma^{OF}_{\text{det}} + w^{OF}_{\text{size}}\sigma^{OF}_{\text{size}}, \tag{2}$$

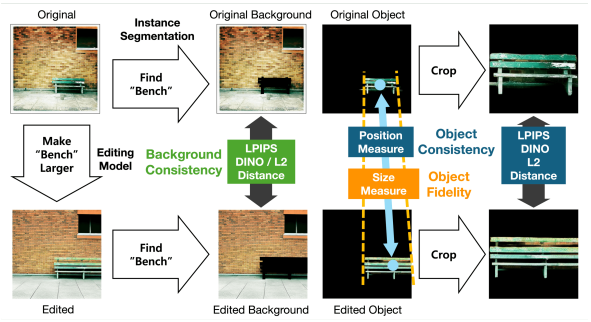
where the weights w^{OF}_{*} are fitted so as to align with human annotations, described in Sec. 3.4.

For Background Fidelity (BF), we similarly measure the CLIP alignment σ^{BF} between the separated background from I_e and the target text C'_e, such as 'living room'.

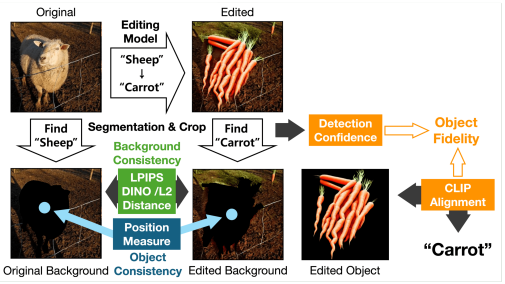
Consistency. Object Consistency (OC) is measured by five scores. We first segment the target object from the original and edited image through the same process as Object Fidelity, and resize them to the same size. Then, we compute the LPIPS distance [35], DINO embedding similarity [23], and L_2 distance, denoted by $\sigma^{OC}_{\text{lpips}}$, $\sigma^{OC}_{\text{dino}}$, and $\sigma^{OC}_{\ell 2}$,



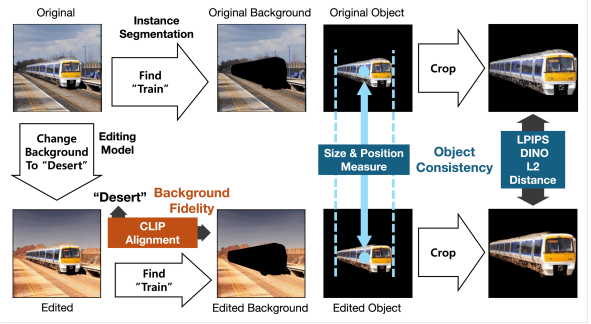
(a) OBJECT ADDITION



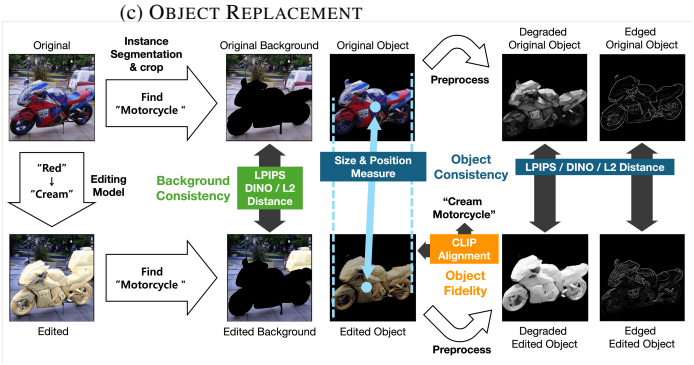
(b) OBJECT RESIZING



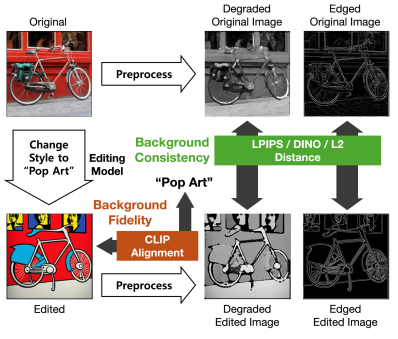
(c) OBJECT REPLACEMENT



(d) BACKGROUND CHANGE



(e) OBJECT ATTRIBUTE CHANGE



(f) STYLE CHANGE

Figure 5. **Evaluation Workflow Specific to Each Editing Task.** See Appendix C.1 for more details.

respectively, between the cropped objects. We also compute position and size consistency, σ^{OC}_{pos} and $\sigma^{OC}_{\text{size}}$, quantifying how much the original position and size of the object is preserved. (See Appendix C.2 for details.) We convert the distance metrics (LPIPS and L_2) to similarity by taking inverse, and normalize all metrics to the same range $[0, 1]$. Similarly, we evaluate Background Consistency (BC) using LPIPS, DINO, and L2 between the original and edited backgrounds, denoted by $\sigma^{BC}_{\text{lpips}}$, $\sigma^{BC}_{\text{dino}}$, and $\sigma^{BC}_{\ell 2}$, respectively.

The overall Object and Background Consistency scores, σ^{OC} and σ^{BC}, are a convex combination of the corresponding metrics:

$$\sigma^{OC} = w^{OC}_{\text{lpips}}\sigma^{OC}_{\text{lpips}} + w^{OC}_{\text{dino}}\sigma^{OC}_{\text{dino}} + w^{OC}_{\ell 2}\sigma^{OC}_{\ell 2} + w^{OC}_{\text{pos}}\sigma^{OC}_{\text{pos}} + w^{OC}_{\text{size}}\sigma^{OC}_{\text{size}}, \quad (3)$$

$$\sigma^{BC} = w^{BC}_{\text{lpips}}\sigma^{BC}_{\text{lpips}} + w^{BC}_{\text{dino}}\sigma^{BC}_{\text{dino}} + w^{BC}_{\ell 2}\sigma^{BC}_{\ell 2}. \quad (4)$$

where the weights w^{OC}_{*} and w^{BC}_{*} are learned by user study described in Sec. 3.4.

Total Score. We aggregate the five scores into a single metric to derive an overall score of a model. For the fidelity and consistency scores, which are measured for each single output image, we take the average over all test cases. Image Quality is computed once from the 'set' of the edited outputs. We aggregate all five scores into a single Total Score σ^{Total} by convex combination:

$$\sigma^{\text{Total}} = \sum_{x \in \mathcal{X}} w^x \sigma^x \quad (5)$$

where $\mathcal{X} = \{\text{IQ, OF, BF, OC, BC}\}$, and the weights are fitted to align with human perception in Sec. 3.4.

Some scoring criteria can be inappropriate for specific tasks; *e.g.*, evaluating Object Consistency (OC) for OBJECT REPLACEMENT task is a non-sense. Thus, each task is evaluated only with a pre-defined set of scoring criteria. Fig. 5 illustrates properly curated workflow to evaluate each task.

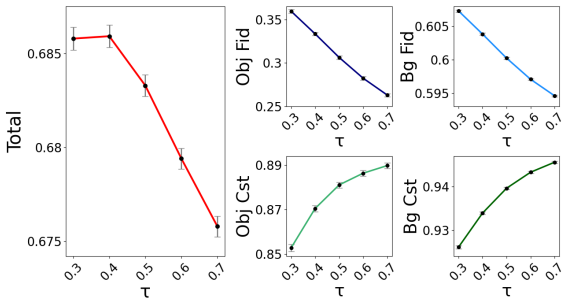


Figure 6. **Aggregated HATIE scores with Varied Editing Intensity for Prompt-to-Prompt.** Larger τ means a weaker edit.

3.4. Human-aligned Benchmark

In order to align the scores from our evaluation framework with human perception, we conduct a user study to evaluate editing results from several models using the same criteria introduced in Sec. 3.3 (except for Image Quality). Then, we calibrate the weights to combine metrics so as to agree with this user study results.

User Study Design. We sample 4,050 output images edited by 6 description-based models (see Sec. 5) for 2,025 editing queries , $\mathcal{M} = \{m_i\}$, evenly across query types. Among these, 2,700 images from 1,350 queries are used for this alignment, leaving the rest for test purpose. These samples are placed into 16 same-sized question sets.

We construct each question to compare edited images by two anonymous models, showing the original image and query as well. Among the 16 sets, 8 of them ask to judge the overall editing quality, while the rest 8 sets ask one of the four specific criteria (OF, BF, OC, BC; 2 sets per each). We divide 24 participants into 8 groups, and assign one question set from overall quality assessment and one set from specific criterion per group. Each question is answered by 3 participants, and we take the majority vote. Appendix C.4 provides examples of the survey questions.

Fitting the Score Weights. For each evaluation criterion $k \in \{OF, BF, OC, BC\}$, we compute the winning rate of each model from the user study result, denoted by $\mathbf{u}^k \in [0,1]^M$, where M is the number of competing models. Suppose now we similarly compute the winning rate using our metrics in Eq. (2)-(4), denoted by $\mathbf{v}^k \in [0,1]^M$. We choose the weights in Eq. (2)-(4) such that the Pearson correlation between $\mathbf{u}^k$ and $\mathbf{v}^k$ is maximized, indicating that our evaluation result is best-aligned with the human perception revealed in the user study. For this, we performed grid search for all possible combinations of weights with step size of 0.01.

The weights to aggregate the five main criteria scores in Eq. (5) are also determined in a similar way. That is, we determine the weights such that the winning rate vector made by user study and that by our method are mostly aligned in terms of Pearson correlation. The degree of fit with user study is plotted in Appendix C.3. We evaluate this alignment using set-aside user study results in Sec. 4.2.

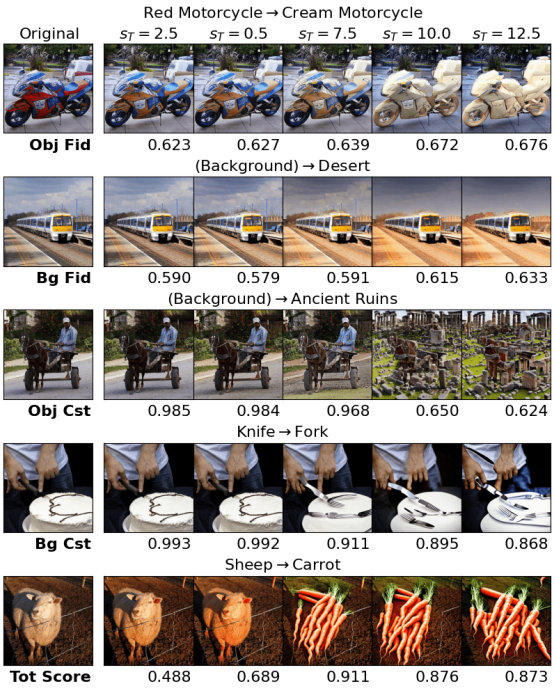


Figure 7. **Demonstration of Our Evaluation Metrics for sample images for each criteria.** Tested model is InstructPix2Pix with $s_T \in \{2.5, 5.0, 7.5, 10.0, 12.5\}$. Higher s_T means stronger edit.

4. Empirical Verification

We first verify the effectiveness of our proposed benchmark and evaluation framework in two ways: quantitative and qualitative demonstration when the intensity of editing changes (Sec. 4.1) and verification of alignment to the human perception (Sec. 4.2).

4.1. Benchmark with Varied Edit Intensity

To verify our benchmark accurately responds to the model characteristics and performance changes, we vary the hyperparameters controlling the degree of edits in several models; specifically, the interpolation parameter η (Imagic [15]), text guidance scale s_T (MasaCtrl [5], FreeDiff [31], and InstructPix2Pix (IP2P) [3]), and self-replace step τ (Prompt-to-Prompt (P2P) with null-text inversion [21]). More details are provided in Appendix D.3.

Quantitative Analysis. Fig. 6 shows the change of aggregated benchmark scores of P2P model as its parameter τ changes (lower τ means stronger edit). As τ increases (the intended edit intensity decreases), two fidelity scores decrease while two consistency scores increase. The total scores show trade-off between fidelity and consistency, and the optimal setting is found where the two aspects are balanced (τ = 0.4). From this result, we conclude that our HATIE precisely captures the model's editing perfor-

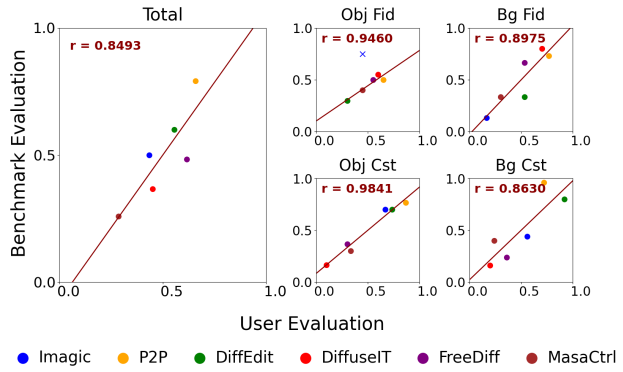


Figure 8. **Relation between winning rates by users and HATIE,** measured on the user study test set. The least square linear fit (red line) and Pearson's correlation coefficient (r) are reported. The cross mark in object fidelity figure exhibits an outlier which has been excluded from correlation calculation.

mance comprehensively, as well as across each specific aspect. It is also notable that the error of each point, measured by bootstrapping, is small, indicating not only that HATIE benchmark is stable, but also that our benchmark is sensitive enough to distinguish the performance gap between the models. This implies that our benchmark is suitable for fine-grained comparison of the models. A similar trend is observed with other models, reported in Appendix D.3.

More practically, as our framework can analyze throughout hyperparameter space, it would be also more reliable for finding the optimal hyperparameters of model than manual exploration.

Qualitative Demonstration. Fig. 7 illustrates evaluation samples with IP2P across different text guidance scales s_T. As the edit intensity gets stronger (higher s_T), model well reflects the intended edit more clearly and lose its original context to some extent. Our fidelity and consistency scores successfully reflect this trend, and the aggregated Total score clearly distinguishes successful and failed edits. This demonstrates that our HATIE cognitively follows the trend of editing quality in each field.

4.2. Alignment to Human Perception

We evaluate the correlation between our benchmark scores and human evaluation on a set-aside user study results which are not used to fit the weights in Sec. 3.4. Among the collected user annotations for 4,050 images from 2,025 edit queries in user study, 1,350 images for 675 queries were left for test purpose, as mentioned in Sec. 3.4. These are split into 4 question sets for overall quality assessment and 4 sets for the specific criteria (1 set for each).

We obtain the correlation between the model winning rates by user study and those by HATIE metrics, in a similar manner to Sec. 3.4, plotted in Fig. 8. Results from ours are successfully aligned with human evaluation results with strong correlation to each other (Pearson's coefficient

Correlation	Object Fidelity	Background Fidelity	Object Consistency	Background Consistency	Total Score
ρ	0.8208	0.8971	0.9276	0.8857	0.7143
τ	0.7379	0.7857	0.8281	0.7333	0.6000

Table 2. **Correlation coefficients between model winning rates from user study and our metrics** on the user study test set.

Metric	r	ρ	τ
CLIP Alignment	-0.3410	-0.2000	-0.2000
Detection Rate	0.4293	0.5218	0.4140
LPIPS	0.5468	0.4857	0.3333
DINO	0.3367	0.0286	0.0667
L_2	0.4599	0.2000	0.0667
Position Score	-0.1497	-0.0580	0.0000
Size Score	-0.3900	-0.3714	-0.2000
Detection Confidence	0.0287	-0.0857	-0.0667
FID	0.5058	0.3714	0.3333

Table 3. **Correlation coefficients between model winning rates from user study and conventional metrics** on overall edit quality assessments in test set survey.

r over 0.8) throughout every criterion. Two other types of correlation coefficients (Spearman's ρ and Kendall's τ) are reported in Tab. 2, showing a similar trend. Overall, our benchmark is well-aligned with user survey results, providing reliable evaluations close to human perception. We also provide alignment result tested on unseen new dataset in Appendix D.4

Comparison with Conventional Metrics. We compare our benchmark's human alignment results with two conventional metrics used in previous studies: (1) CLIP alignment, which measures the similarity between the target caption (C_e) and the edited image (I_e), as used in [10, 15] , and (2) detection rate used in [2] which evaluates the presence of objects generated by the queries, 1 if object is detected and 0 if not. We also compare our results to each metrics used in HATIE. Within HATIE framework, instead of aggregating multiple metric scores, we use each metric alone and compare the results with the user evaluation.

Tab. 3 shows the correlation coefficients between user evaluation and conventional metrics on the overall edit quality in user study test set in the upper section, and same data obtained with single metrics in the lower section. This clearly shows much inferior alignment with human evaluation results, compared to ours in Tab. 2. Our structured and combined use of conventional metrics enables us to successfully adjust benchmarks to better align with human perceptions.

5. HATIE Benchmark Results and Discussion

In this section, we evaluate image editing models using the proposed framework. We compare 6 description-based editing models (Imagic, P2P, DiffEdit [6], DiffuseIT [16], FreeDiff, and MasaCtrl) and 3 instruction-based models (IP2P, InstructDiffusion (InstDiff) [9], and MagicBrush [34]). For models with adjustable parameters, we

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	0.3174 ± 0.0020	0.5964 ± 0.0001	0.8631 ± 0.0013	0.9150 ± 0.0002	0.7180 ± 0.0014	0.6739 ± 0.0006
	P2P	**0.3595 ± 0.0020**	**0.6073 ± 0.0002**	0.8528 ± 0.0015	0.9261 ± 0.0003	0.6551 ± 0.0023	**0.6858 ± 0.0006**
	Imagic	0.2271 ± 0.0017	0.5878 ± 0.0001	**0.9237 ± 0.0009**	0.9201 ± 0.0003	**0.8406 ± 0.0014**	0.6682 ± 0.0005
	MasaCtrl	0.2332 ± 0.0019	0.5967 ± 0.0001	0.6754 ± 0.0023	0.8954 ± 0.0003	0.2685 ± 0.0022	0.5936 ± 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	0.4474 ± 0.0021	0.6169 ± 0.0002	0.7903 ± 0.0021	0.9319 ± 0.0003	0.6658 ± 0.0031	0.6960 ± 0.0007

Table 4. HATIE benchmark results across score criteria

Models		Object Addition	Object Replacement	Object Resizing	Attribute Change	Background Change	Style Change
Description -based	DiffEdit	0.5512 ± 0.0013	0.5190 ± 0.0012	0.6666 ± 0.0016	0.7687 ± 0.0014	0.7382 ± 0.0012	**0.7517 ± 0.0003**
	DiffuseIT	0.5452 ± 0.0016	0.5691 ± 0.0020	0.6780 ± 0.0015	0.7919 ± 0.0009	0.7319 ± 0.0010	0.7208 ± 0.0003
	FreeDiff	0.5101 ± 0.0013	0.5759 ± 0.0022	**0.7054 ± 0.0016**	0.7896 ± 0.0010	0.7327 ± 0.0011	0.7411 ± 0.0003
	P2P	**0.6246 ± 0.0020**	**0.6088 ± 0.0023**	0.6899 ± 0.0015	0.8031 ± 0.0010	0.7111 ± 0.0015	0.7374 ± 0.0004
	Imagic	0.4964 ± 0.0010	0.4689 ± 0.0010	0.6863 ± 0.0012	**0.8244 ± 0.0006**	**0.7585 ± 0.0008**	0.7304 ± 0.0003
	MasaCtrl	0.4829 ± 0.0011	0.4804 ± 0.0012	0.6388 ± 0.0020	0.6739 ± 0.0017	0.6376 ± 0.0020	0.7351 ± 0.0003
Instruction -based	MagicBrush	**0.7461 ± 0.0018**	**0.8189 ± 0.0023**	0.7279 ± 0.0016	**0.8407 ± 0.0005**	**0.6536 ± 0.0023**	**0.7471 ± 0.0004**
	InstDiff	0.6260 ± 0.0020	0.6656 ± 0.0025	**0.7488 ± 0.0019**	0.7982 ± 0.0009	0.5206 ± 0.0024	0.7066 ± 0.0005
	IP2P	0.6771 ± 0.0021	0.7511 ± 0.0025	0.6929 ± 0.0012	0.8295 ± 0.0007	0.5800 ± 0.0024	0.7407 ± 0.0004

Table 5. HATIE benchmark results across various edit types

Models		Whole Image	Person	Animal	Vehicle	Household	Dining	Outdoor	Other
Description -based	DiffEdit	0.7963 ± 0.0008	0.7087 ± 0.0026	0.6506 ± 0.0026	0.6980 ± 0.0022	0.6936 ± 0.0021	0.6344 ± 0.0031	0.6556 ± 0.0026	0.6618 ± 0.0029
	DiffuseIT	0.7669 ± 0.0007	0.7090 ± 0.0024	0.7293 ± 0.0017	0.7141 ± 0.0019	0.6734 ± 0.0021	0.6735 ± 0.0024	0.6909 ± 0.0022	0.6817 ± 0.0024
	FreeDiff	0.7837 ± 0.0007	**0.7208 ± 0.0026**	0.7268 ± 0.0019	0.7175 ± 0.0021	0.6894 ± 0.0022	0.6534 ± 0.0028	0.6951 ± 0.0022	0.6880 ± 0.0026
	P2P	0.7632 ± 0.0010	0.7179 ± 0.0024	**0.7428 ± 0.0021**	**0.7448 ± 0.0020**	**0.7154 ± 0.0022**	**0.6922 ± 0.0029**	**0.7224 ± 0.0021**	**0.7164 ± 0.0026**
	Imagic	**0.7967 ± 0.0006**	0.7085 ± 0.0021	0.7003 ± 0.0016	0.7043 ± 0.0017	0.6977 ± 0.0017	0.6619 ± 0.0021	0.6984 ± 0.0018	0.6819 ± 0.0021
	MasaCtrl	0.7163 ± 0.0013	0.6819 ± 0.0032	0.6197 ± 0.0026	0.6527 ± 0.0027	0.5527 ± 0.0032	0.5507 ± 0.0036	0.5925 ± 0.0029	0.5771 ± 0.0036
Instruction -based	MagicBrush	**0.7273 ± 0.0015**	**0.7613 ± 0.0027**	**0.8254 ± 0.0016**	**0.8162 ± 0.0018**	**0.7841 ± 0.0019**	**0.8084 ± 0.0020**	**0.8064 ± 0.0017**	**0.7935 ± 0.0022**
	InstDiff	0.6114 ± 0.0016	0.7520 ± 0.0031	0.7770 ± 0.0020	0.7488 ± 0.0022	0.6976 ± 0.0023	0.7206 ± 0.0027	0.7560 ± 0.0020	0.7252 ± 0.0025
	IP2P	0.6749 ± 0.0016	0.7214 ± 0.0019	0.7786 ± 0.0017	0.7726 ± 0.0019	0.7635 ± 0.0021	0.7808 ± 0.0022	0.7829 ± 0.0019	0.7689 ± 0.0023

Table 6. HATIE benchmark results across target object classes

choose the default value found in the officially released code ($s_T = 7.5$, $s_I = 1.5$ and $\tau = 0.3$). For Imagic, we choose $\eta = 0.6$, which we achieve the best result.

Tab. 4 shows the benchmark performance of the competing models, both for description-based and instruction-based. Among the description-based models, Imagic and DiffEdit tend to be more competitive in consistency, while DiffuseIT and P2P are more competent in fidelity. P2P turns out to be the best performing all-round, showing outstanding results both for fidelity and consistency. Among the instruction-based models, MagicBrush performs the best in most areas, leading to the highest total score. It is also notable that models with high consistency scores tend to show great image quality score in both description and instruction-based group. (We emphasize again that models using different caption types are not fairly comparable.)

Tab. 5 and 6 show the benchmark scores aggregated separately by edit types and edited object classes. "Whole Image" in Tab. 6 stands for targets of non-object centric edits. From the tables, we can see that each model often has its strong and weak points. For example, looking at Tab. 5, Imagic tends to produce high quality image, performing ATTRIBUTE CHANGE and BACKGROUND CHANGE relatively better, while showing weakness on OBJECT ADDITION or REPLACEMENT. InstDiff underperforms in most edit tasks compared to other instruction-based models, but is particularly strong for OBJECT RESIZING. In Tab. 6, P2P performs well on most of the objects, while it has weakness in whole image scale edits. Oppositely, Imagic underperforms in every target classes except for edits on whole image.

6. Summary and Limitations

We present a novel benchmark HATIE, which includes a vast dataset covering a wide range of editing types and object classes. Our evaluation scheme provides accurate assessment through a structured use of multiple metrics in categorized criteria, quantifying the performance of an editing model in a manner similar to human perception. HATIE allows precise evaluation and analysis of text-guided image editing model, and we hope it will guide the future direction of image editing research.

Limitations. HATIE aims to cover a wide range of editing tasks, but some are excluded. Object removal for description-based models and movement/rotation are omitted due to challenges in generating captions and defining reference points. Certain attribute types are excluded due to difficulties in localization. See Appendix E for more details.

Acknowledgments

This work was supported by Samsung Electronics (IO240512-09881-01), Youlchon Foundation, by SOFT Foundry Institute at SNU, and NRF grants (RS-2021-NR05515, RS-2024-00336576, RS-2023-00222663) and IITP grants (RS-2024-00353131, RS-2022-II220264), funded by the government of Korea.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022. 2

[2] Samyadeep Basu, Mehrdad Saberi, Shweta Bhardwaj, Atoosa Malemir Chegini, Daniela Massiceti, Maziar Sanjabi, Shell Xu Hu, and Soheil Feizi. EditVal: Benchmarking diffusion based text-guided image editing methods. *arXiv:2310.02426*, 2023. 1, 2, 7

[3] Tim Brooks, Aleksander Holynski, and Alexei A Efros. InstructPix2Pix: Learning to follow image editing instructions. In *CVPR*, 2023. 2, 6

[4] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 8(6):679–698, 1986. vi

[5] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. MasaCtrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, 2023. 2, 6

[6] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. DiffEdit: Diffusion-based semantic image editing with mask guidance. In *ICLR*, 2023. 2, 7

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. xi

[8] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv:2407.21783*, 2024. 4, ii

[9] Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Houqiang Li, Han Hu, et al. InstructDiffusion: A generalist modeling interface for vision tasks. In *CVPR*, 2024. 1, 2, 7

[10] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2023. 1, 2, 7

[11] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning". In *EMNLP*, 2021. 4

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *NIPS*, 2017. 4

[13] Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, et al. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *CVPR*, 2024. 1

[14] Drew A Hudson and Christopher D Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. 2

[15] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, 2023. 1, 2, 6, 7

[16] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. In *ICLR*, 2023. 2, 7

[17] Chenliang Li, Haiyang Xu, Junfeng Tian, Wei Wang, Ming Yan, Bin Bi, Jiabo Ye, Hehong Chen, Guohai Xu, Zheng Cao, et al. mPLUG: Effective and efficient vision-language learning by cross-modal skip-connections. In *EMNLP*, 2022. i

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 3

[19] Yuanze Lin, Yi-Wen Chen, Yi-Hsuan Tsai, Lu Jiang, and Ming-Hsuan Yang. Text-driven image editing via learnable regions. In *CVPR*, 2024. 2

[20] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2022. 1

[21] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *CVPR*, 2023. 6

[22] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. 2

[23] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. In *TMLR*, 2023. 4

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[25] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 1

[26] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[27] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022. 1

[28] Jing Shi, Ning Xu, Trung Bui, Franck Dernoncourt, Zheng Wen, and Chenliang Xu. A benchmark and baseline for language-driven image editing. In *ACCV*, 2020. 1, 2

[29] Qian Wang, Biao Zhang, Michael Birsak, and Peter Wonka. InstructEdit: Improving automatic masks for diffusion-based image editing with user instructions. *arXiv:2305.18047*, 2023. 2

[30] Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J Fleet, Radu Soricut, et al. Imagen editor and EditBench: Advancing and evaluating text-guided image inpainting. In *CVPR*, 2023. 1, 2

[31] Wei Wu, Qingnan Fan, Shuai Qin, Hong Gu, Ruoyu Zhao, and Antoni B Chan. FreeDiff: Progressive frequency truncation for image editing with diffusion models. In *ECCV*, 2024. 2, 6

[32] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 3, 4

[33] Yiqi Wu, Xiaodan Hu, Ziming Fu, Siling Zhou, and Jiangong Li. GPT-4o: Visual perception performance of multimodal large language models in piglet activity understanding. *arXiv:2406.09781*, 2024. 4, ii

[34] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. *NeurIPS*, 2024. 1, 2, 7

[35] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 1, 4