Click2Mask: Local Editing with Dynamic Mask Generation

Omer Regev, Omri Avrahami, Dani Lischinski

The Hebrew University of Jerusalem

Abstract

Recent advancements in generative models have revolutionized image generation and editing, making these tasks accessible to non-experts. This paper focuses on local image editing, particularly the task of adding new content to a loosely specified area. Existing methods often require a precise mask or a detailed description of the location, which can be cumbersome and prone to errors. We propose Click2Mask, a novel approach that simplifies the local editing process by requiring only a single point of reference (in addition to the content description). A mask is dynamically grown around this point during a Blended Latent Diffusion (BLD) process, guided by a masked CLIP-based semantic loss. Click2Mask surpasses the limitations of segmentation-based and fine-tuning dependent methods, offering a more user-friendly and contextually accurate solution. Our experiments demonstrate that Click2Mask not only minimizes user effort but also enables competitive or superior local image manipulations compared to SoTA methods, according to both human judgement and automatic metrics. Key contributions include the simplification of user input, the ability to freely add objects unconstrained by existing segments, and the integration potential of our dynamic mask approach within other editing methods.

Project Page & Code —
https://omeregev.github.io/click2mask

Full Paper — https://arxiv.org/abs/2409.08272

1 Introduction

Recent advances in generative models have revolutionized image generation and editing capabilities, enabling both streamlined workflows and accessibility for non-experts. The latest approaches utilize natural language to manipulate images either globally – altering the content or style of the entire image – or locally – adding, removing, or modifying specific objects within a limited image region.

In this work, we focus on local editing, specifically on the task of adding new content in a local area. Similar to DragDiffusion (Shi et al. 2023) for movement and MagicEraser (Li et al. 2024) for removal, this focused scope leverages specialization to tackle the unique challenges of local editing. To accomplish such edits, some existing methods require users to provide explicit precise masks (Avrahami, Lischinski, and Fried 2022; Ramesh et al. 2022; Avrahami, Fried, and Lischinski 2023; Wang et al. 2023b; Xie et al. 2022), which is tedious and may yield unexpected results due to lack of mask precision. Other methods describe the desired manipulations in natural language, as an edit instruction (Brooks, Holynski, and Efros 2023; Sheynin et al. 2023), or by providing a caption and the desired change (Bar-Tal et al. 2022; Kawar et al. 2023; Hertz et al. 2022; Tumanyan et al. 2022). These methods also require user expertise, and their results may suffer from ambiguous or imprecise prompts. Moreover, they fail to ensure that the changes to the image are confined to a local area, or that they occur at all, as demonstrated in Figure 1.

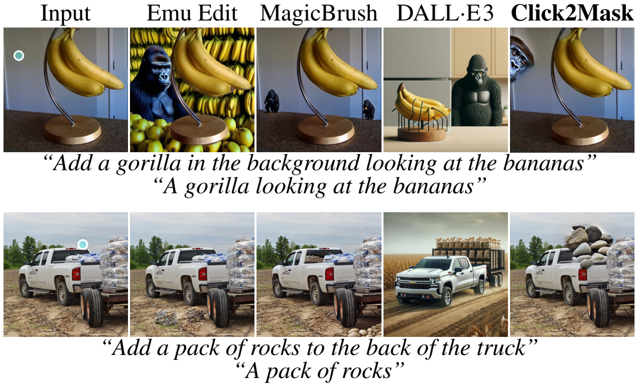

Input
Emu Edit
MagicBrush
DALL·E3
Click2Mask
"Add a gorilla in the background looking at the bananas"
"A gorilla looking at the bananas"
"Add a pack of rocks to the back of the truck"
"A pack of rocks"

Figure 1: **Comparisons to SoTA models.** A comparison of Emu Edit (Sheynin et al. 2023), MagicBrush (Zhang et al. 2023) and DALL·E 3 (Betker et al. 2023) with our model **Click2Mask**. In each example, the top prompt was given to the other models, while Click2Mask received the simpler bottom prompt, in addition to the blue dot (mouse click) on the input. Other models completely change the image, or the background, fail to edit, or produce unrealistic results.

To overcome the aforementioned shortcomings, we introduce Click2Mask, a novel approach that simplifies user interaction by requiring only a single point of reference rather than a detailed mask or a description of the target area. The provided point gives rise to a mask that dynamically

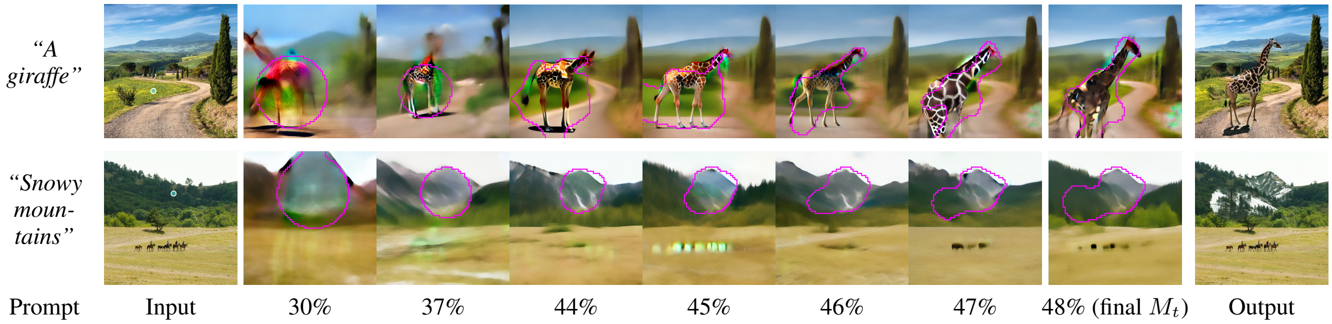


Figure 2: **Mask evolution**. A visualization of the mask evolution throughout the diffusion process. Leftmost image is input with clicked point, rightmost image is the final Click2Mask output. Intermediate images are decoded latents $\tilde{z}_{fg}$ at several diffusion steps, where the purple outline depicts the contour of current (upscaled) mask M_t. Percentages indicate the step out of 100 diffusion steps, with the last being the final evolved mask.

evolves through a Blended Latent Diffusion (BLD) process (Avrahami, Lischinski, and Fried 2022; Avrahami, Fried, and Lischinski 2023), where the evolution is guided by a semantic loss based on Alpha-CLIP (Sun et al. 2023). This process (Figure 2) enables local edits that are both precise and contextually relevant (Figures 1 and 3).

Unlike segmentation-based methods that depend on preexisting objects (Couairon et al. 2022; Xie et al. 2023; Wang et al. 2023a; Zou et al. 2024), Click2Mask does not confine the edit area to the boundaries of an existing segment. Furthermore, in contrast to editing approaches that require fine-tuning the diffusion model (Wang et al. 2023b; Xie et al. 2022; Kawar et al. 2023; Avrahami et al. 2023), we employ pre-trained models, and only perform context dependent optimization on the mask.

Our experiments demonstrate that Click2Mask not only reduces the effort required by users but also achieves competitive or superior results compared to state-of-the-art methods in local image manipulation.

In summary, our contributions are: (i) Reduction of user effort by eliminating the need for precise mask outlines, or overly descriptive prompts. (ii) Ability to add objects in a free-form manner, unconstrained by boundaries of existing objects or segments. (iii) Our dynamically evolving mask approach is not a stand-alone method, but rather it can be embedded as a mask generation of the fine-tuning step within other methods that internally employ a mask, such as Emu Edit (Sheynin et al. 2023) which currently generates multiple masks (a precise mask using DINO (Caron et al. 2021) and SAM (Kirillov et al. 2023), an expanded version of it, and a bounding box), and filters the best result from multiple images produced using these masks.

2 Related Work

In recent years, much work has been done on image generation, with diffusion-based models (DMs) (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020; Dhariwal and Nichol 2021; Rombach et al. 2022; Ramesh et al. 2022; Saharia et al. 2022) facilitating a host of SoTA text-guided image editing methods and capabilities.

Mask-based approaches. Text-guided image manipulation may naturally be limited to a specific region using a mask. In the context of DMs this was first explored in Blended Diffusion (Avrahami, Lischinski, and Fried 2022), where a user-provided mask is used to blend images throughout a denoising process with a text-guided noisy image. This approach was later incorporated into Latent Diffusion (Rombach et al. 2022) by performing the blending in latent space. The resulting Blended Latent Diffusion (BLD) method (Avrahami, Fried, and Lischinski 2023) serves as the basis for our work and described in more detail in Section 3. GLIDE (Nichol et al. 2022), Imagen Editor (Wang et al. 2023b) and SmartBrush (Xie et al. 2022) fine-tuned the DM for image inpainting, by obscured training images or by conditioning on a mask. However, user-provided masks have a major disadvantage: the success of the edit depends on the exact shape of the mask, which can be tedious and time-consuming for a user to create.

Mask-free approaches. Both Text2Live (Bar-Tal et al. 2022), which generates a composite layer, and Imagic (Kawar et al. 2023), which interpolates target text and optimized source embeddings, fine-tune the generative model for each image, which is quite costly, contrary to our work. Several works use attention injection, such as Plug-and-Play (Tumanyan et al. 2022) and Prompt-to-Prompt (Hertz et al. 2022), where the latter requires a time-consuming caption of the input image, unlike our method. Most of these methods focus on altering a certain object (by replacement, removal or style change), or applying global changes (style or content), in contrast to our focus on adding objects freely.

Instruction-based approaches. Other methods can add objects in a free manner. InstructPix2Pix (Brooks, Holynski, and Efros 2023) (subsequently fine-tuned by MagicBrush (Zhang et al. 2023)) produces (instruction, image) pairs, used to train an instruction-conditioned DM. Emu Edit (Sheynin et al. 2023) is a more recent model trained on a wide range of learned task embeddings to enable instruction-based image editing, however, it is not publicly available. DALL·E3 (Betker et al. 2023) is also proprietary, and modifies the entire image as demonstrated in Figure 1. DALL·E3 and DALL·E 2 (Ramesh et al. 2022) apparently support masked inpainting, but we are unaware of a publicly avail-

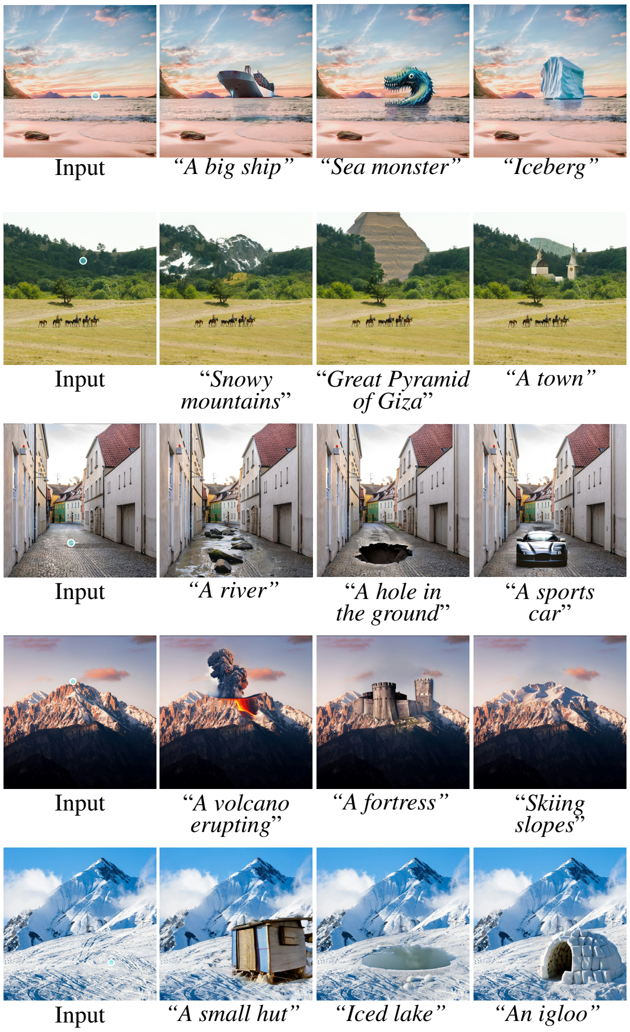


Figure 3: **Examples of Click2Mask outputs.** The leftmost column is the input image with clicked point. The other columns are **Click2Mask** outputs given the prompts below.

able way to apply it to general real images. MGIE (Fu et al. 2024) train a DM, utilizing a MLLM to derive expressive instructions. These methods require the user to specify the desired localization in words, which has a few shortcomings. On the user's side, this requires effort, and it can be difficult or impossible to describe the precise location. From the model's side, failure to visually ground the text-specified location may fail to perform the desired edit, and/or make unintended changes in other locations instead.

Segmentation-based approaches. Segmentation methods have been utilized to overcome the need for a precise user-provided mask. DiffEdit (Couairon et al. 2022) and Edit Everything (Xie et al. 2023) generate segmentation-based masks by utilizing conditionings on diffusion steps, or SAM (Kirillov et al. 2023), but require an input image caption, which is painstaking. InstructEdit (Wang et al. 2023a), which uses Grounding DINO (Liu et al. 2023) and SAM to generate a mask, does not require one, but requires a description of the object to alter. This can cause errors due to failure of the model to localize. InstDiffEdit (Zou et al. 2024) generates masks based on attention maps during denoising.

The segmentation-based methods, however, suffer from a few limitations: (i) Such models need to "lock" on an existing object or segment; consequently, in most cases they alter objects, but do not add new free-form ones, which is our focus. (ii) These methods typically require the user to provide an input caption or a description of the altered object.

In contrast to all the above, our work enables *adding* objects to real images (as opposed to merely altering existing ones), without having to provide a precise mask, to describe the input image, or target image, and without being constrained to boundaries of existing objects or segments. We aim to enable edits where the manipulated area is not well-defined in advance, and a free-form alteration is required.

3 Blended Latent Diffusion

Blended Latent Diffusion (BLD) (Avrahami, Fried, and Lischinski 2023) is a method for local text-guided image manipulation, based on Latent Diffusion Models (LDMs) (Rombach et al. 2022) and Blended Diffusion (Avrahami, Lischinski, and Fried 2022). Given a source image x, a guiding text prompt p, and a binary mask m, the model blends the source latents (obtained by DDIM inversion (Song, Meng, and Ermon 2020)) with the prompt-guided latents throughout the LDM process, to derive a blended final output.

Initially, inputs are converted to a latent space. A variational auto-encoder (Kingma and Welling 2013) with encoder $E(\cdot)$ and decoder $D(\cdot)$, encodes x to latent space, s.t. $z_{init} = E(x)$. In addition, m is downsampled to m_{latent} in order to meet latent spatial dimensions.

In each BLD step t, the following occurs:

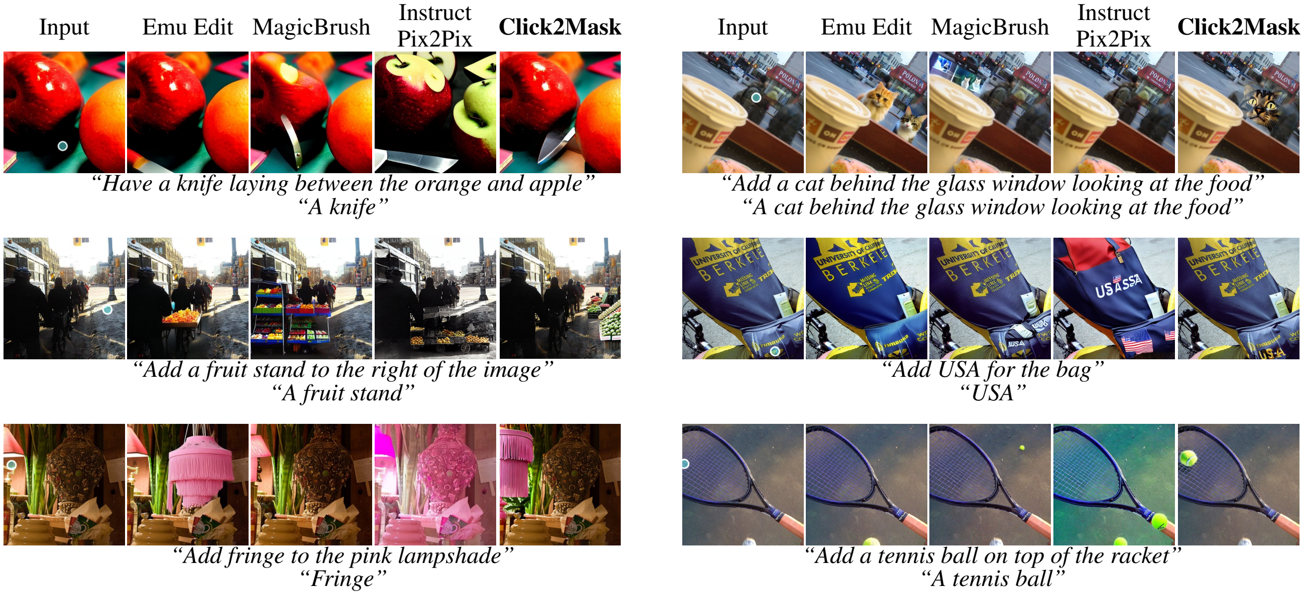
1. The latent resulting from the previous step, z_{t+1}, undergoes denoising conditioned by the prompt p, to yield z_{fg} (we refer to the generated content as *foreground*, or *fg*).
2. The original image latent z_{init} is noised to step t, yielding z_{bg} (we refer to the original content as *background*, *bg*).
3. The next step z_t is obtained by blending z_{fg} and z_{bg} using m_{latent}:

$$z_t = z_{fg} \odot m_{latent} + z_{bg} \odot (1 - m_{latent}) \tag{1}$$

where $\odot$ denotes element-wise multiplication.

After the final step, the output z_0 is decoded to obtain the final edited image $\hat{x} = D(z_0)$.

However, because information is lost during the VAE encoding, the decoded final output $\hat{x}$, might exhibit some artifacts when the unmasked region has important fine-detailed content (such as faces, text, etc.). Avrahami et al. (2023) solve this issue by optionally fine-tuning the decoder weights for each image after the denoising steps, and using these weights to infer the final result. In our experiments, we found that this optional background preservation process is no longer necessary (possibly due to improvements in the Stable Diffusion VAE), and a final blending with Gaussian feathering suffices (refer to Figure 13 in appendix; full paper link available in the Footnote in Page 1).



"Have a knife laying between the orange and apple"
"A knife"

"Add a cat behind the glass window looking at the food"
"A cat behind the glass window looking at the food"

"Add a fruit stand to the right of the image"
"A fruit stand"

"Add USA for the bag"
"USA"

"Add fringe to the pink lampshade"
"Fringe"

"Add a tennis ball on top of the racket"
"A tennis ball"

Figure 4: **Comparisons with SoTA methods.** Comparisons of Emu Edit (Sheynin et al. 2023), MagicBrush (Zhang et al. 2023) and InstructPix2Pix (Brooks, Holynski, and Efros 2023) with our model **Click2Mask**. Upper prompts were given to baselines, and lower ones to Click2Mask. The inputs contain the clicked point given to Click2Mask. As Figure 8 shows, baselines often modify unrelated objects, make global changes, misplace elements, or replace rather than add objects. See appendix for more comparisons.

4 Method

Given an image, a text prompt, and a user-indicated location (e.g., via a mouse click), our goal is to modify the image according to the prompt in an unspecified area roughly surrounding the provided point. We utilize Blended Latent Diffusion (BLD) (Avrahami, Fried, and Lischinski 2023) as our image editing backbone, but rather than providing it with a fixed mask at the outset, we evolve a mask dynamically throughout the diffusion process. We initialize the process with a large mask around the indicated point, and gradually *contract* the mask towards the center, while guiding the rate of contraction along the mask boundary using a semantic alignment loss based on Alpha-CLIP (Sun et al. 2023).

This iterative process results in a mask whose shape and size are determined by both the text prompt, the content, and the structure of the original input image. Furthermore, the shape of the mask adjusts itself to the emerging object, as the mask's evolution is determined by the gradients obtained by the semantic alignment loss (see Section 4.1), which in turn depend on the shape of the object being generated (see Figure 2 for mask evolution illustration, and Figure 5 for examples of generated masks). Once the mask has settled into its final form, we run BLD once more, using the final mask to generate the final result. Our method is outlined in Algorithm 1 and illustrated in Figure 6.

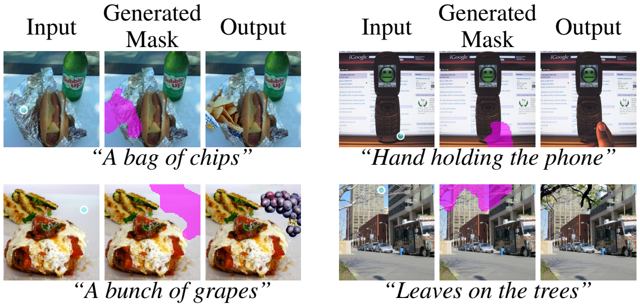
4.1 Dynamic Mask Evolution

Given an image x, a text prompt p, and a user-provided location c, we aim to modify x, so as to align with p, in proximity to c. We start by encoding the input image $z_{init} = E(x)$. We also create a 2D potential height-field Φ in latent space, which is initialized to a Gaussian around c.

We now perform the BLD process, where at each step t we obtain a binary mask M_t by thresholding the potential Φ using a threshold τ. The mask evolves dynamically through the BLD process, since the threshold τ and the potential Φ are both updated at each step: the threshold τ increases, while the potential Φ is elevated — starting from a specific step, as explained later — in important areas to ensure they remain above the threshold. This prevents the mask from shrinking in spatial areas that emerge as important for alignment of the generated new content with the guiding prompt p. As a consequence, the mask evolves into a shape determined by the newly generated object.

Commencing the blending at 25% of the diffusion steps, the initial threshold value τ_{init} is relatively low, such that M_t is sufficiently large at the beginning (∼16% of the image). This enables BLD to capture the desired edit, as demonstrated in Figure 7 (this idea was originally introduced in BLD to cope with the case of small or thin input masks). On the other hand, to prevent overly large masks that could result in large-scale changes failing to blend seamlessly with the original content, we increase τ rapidly at the beginning, and delay first potential elevation step (denoted b) to 40% of the total diffusion steps. This ensures potential elevation starts late enough to control mask size but still early enough, when the blended image is noisy and can be modified. We stop mask evolution when the spatial structure is nearly determined (at 50% of the total diffusion steps, denoted l).

The potential elevation is obtained by generating the estimated final image $\tilde{x}_0$ at each step, and calculating the co-



"A bag of chips"

"Hand holding the phone"

"A bunch of grapes"

"Leaves on the trees"

Figure 5: **Examples of generated masks.** For each triplet, given an input image with clicked point (left) and a prompt (below), a purple overlay shows the generated mask (middle). The rightmost image is Click2Mask output.

sine distance between the CLIP (Radford et al. 2021) embeddings of $\tilde{x}_0$ and the guidance prompt p. $\tilde{x}_0$ is obtained by blending a predicted final foreground latent $\tilde{z}_{fg}$, with the original latent background z_{init}:

$$\tilde{z}_0 = \tilde{z}_{fg} \odot M_t + z_{init} \odot (1 - M_t) \qquad (2)$$

The decoded $\tilde{x}_0 = D(\tilde{z}_0)$ is passed alongside the current mask M_t and the prompt p to Alpha-CLIP to focus on the area of M_t. The gradient of the cosine distance with respect to the latent mask pixels is then calculated by backpropagating through the CLIP embeddings and the decoder. The larger the absolute gradient of the cosine distance (i.e. CLIP loss) with respect to a pixel in M_t, the more significant this location is for the alignment of the generated content to the prompt p. Adding the absolute gradient values G to Φ, elevates important areas in the Φ height-field (around M_t's contour for stable evolution – Figures 14 and 15 in appendix).

Halfway through the mask evolution steps (denoted k), we initiate an optional stoppage of M_t's evolution if the Alpha-CLIP loss does not decrease in subsequent steps.

Starting from the first Φ elevation step b, after each update of M_t, we restart the BLD process, letting it proceed from the beginning to the current step t, using the mask M_t as a fixed mask. This is to allow pixels that were added (or removed) in M_t to affect the generated image from the beginning (see Figure 16 in appendix).

We then apply Equation (1) and blend z_{fg} with z_{bg} using the mask M_t, which provides z_{t-1}, the input to next step.

After all mask evolution steps have been completed, we perform a final BLD run using the final M_t with several seeds to obtain several candidate results, where the best one is filtered by Alpha-CLIP. As noted earlier, rather than fine-tuning the VAE decoder weights to preserve the original background details outside the mask, we employ instead a simple Gaussian mask feathering when blending the BLD output and the original input image (in pixel space).

5 Results

Given that our method is mask-free, we compare ourselves to mask-free image editing methods, with the slight difference being that a *clicked point* replaces location-describing text in the prompt. As our paradigm is novel and lacks a directly aligned method for comparison, using a single click instead of detailed text is a reasonable trade-off. To begin with, we compare to MagicBrush, which is the SoTA method among the open-source models. In addition, we compare to Emu Edit, which is the SoTA among closed-source models. Since we are unable to run Emu Edit ourselves, we must rely on the Emu Edit Benchmark (Sheynin et al. 2023), which includes images generated by Emu Edit. This benchmark contains images with several categories of editing instructions, such as adding objects, removing objects, altering style, etc. As our focus is adding objects to images, we filtered the dataset by the "addition" instruction. This resulted in 533 items, from which we randomly sampled an evaluation subset of 100 samples.

Algorithm 1: **Click2Mask**

```
**Given:** **models** $LDM = \{noise(z,t), denoise(z,p,t) \rightarrow (z_t, z_0)\}$, $VAE = \{E(x), D(z)\}$, $BLD = \{(x,p,m,t) \rightarrow z_t\}$, $Alpha\text{-}CLIP = \{\alpha_{CLIP}(x,m,p) \rightarrow Sim_{CLIP}\}$, and **hyper parameters** $\{\tau_{n \ldots l}, lr\}$ with schedulers $\{n, b, k, l\}$

**Input:** input image $x$, text prompt $p$, target coordinates $c$
**Output:** edited image $\widehat{x}$ that matches the prompt $p$ in proximity of $c$, and complies to $x$ outside edited region

$\Phi = Gaussian(c)$
$z_{init} = E(x)$
$z_n \sim noise(z_{init}, n)$
**for** all $t$ from $n$ to $l$ **do**
    $z_{bg} \sim noise(z_{init}, t)$
    $z_{fg}, \tilde{z}_{fg} \sim denoise(z_t, p, t)$
    $G = 0$
    **if** $t < b$ **then**
        $\tilde{z}_0 = \tilde{z}_{fg} \odot M_t + z_{init} \odot (1 - M_t)$
        $S_t \sim \alpha_{CLIP}(D(\tilde{z}_0), upscale(M_t), p)$
        $G \sim |gradients(S_t, M_t)|$
        $z_{fg} \sim BLD(x, p, M_t, t)$
    **end if**
    **if** $t < k$ and $S_t > S_{t+1}$ **then exit loop**
    $M_t = (\Phi + G * lr) > \tau_t$
    $z_t = z_{fg} \odot M_t + z_{bg} \odot (1 - M_t)$
**end for**
$\widehat{z} \sim BLD(x, p, M_t, 0)$
**return** $D(\widehat{z})$
```

We perform the following fixed routine for each sample: (i) Removed the word that instructs addition (e.g., "Add", "Insert"), (ii) removed the part that describes the edit location, and instead (iii) clicked on the image to direct the editing location. For instance, the instruction "Add a black baseball cap to the man on the left" becomes "A black baseball cap" (non-localized instruction).

Following Emu Edit (Sheynin et al. 2023) and BLD (Avrahami, Fried, and Lischinski 2023), each sample run produces multiple results internally (3 mask evolutions, each followed by a batch of 8 outputs), and outputs the best result, as determined automatically using Alpha-CLIP scoring.

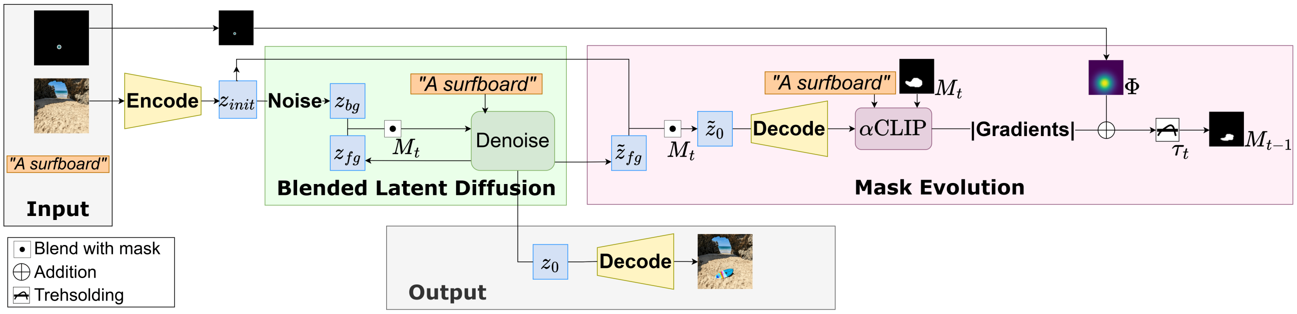


Figure 6: **Click2Mask:** An illustration of our method as described in Algorithm 1. The **green block** is BLD process, performing diffusion steps while blending noised input latents with text guided latents. The **pink block** is the mask evolution process, where we utilize Alpha-CLIP to evaluate the gradients with respect to the mask M_t pixels, using them to update M_t, obtaining M_{t-1}.

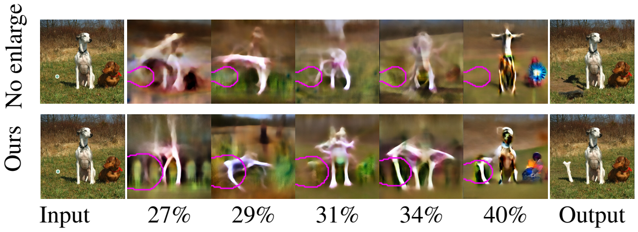


Figure 7: **Ablation study: No early mask enlargement.** As explained in Section 4, we start with a large mask (∼16% of the image), to capture the desired edit in M_t. Top: M_t (purple contours on decoded $\tilde{z}_{fg}$s, throughout diffusion steps indicated by %s) evolves without an initial enlargement, and the diffusion guides the white dog to the prompt *"Huge bone"*, while the small M_t fails to capture the bone. Bottom: Click2Mask's enlarged M_t captures the guided content although the dog is also initially identified as the bone.

To evaluate our results, we compared these 100 outputs generated by Click2Mask, with the outputs generated by Emu Edit and by MagicBrush (which ran with the original edit instructions). We conducted the evaluation through a user study (Section 5.1), as well as through automatic metrics (Section 5.2). In both cases, our method outperformed the SoTA methods.

5.1 Human Evaluation

We conducted a user study, where participants were given a random batch of survey items out of 200 total items (100 items comparing to each model). Each item included an input image, the original edit instruction, and a pair of edited images: one generated by our model, and the other generated by either Emu Edit or MagicBrush. Participants were asked to rank which of the edited images performed better according to three criteria: executing the instruction, not adding any other edits or artifacts, and generating a realistic image. The survey was completed by 149 participants. Each of the 200 items was rated by at least 5 users, where the average rate was 15.67 users in Emu Edit, and 8.06 users in MagicBrush.

In order to compare Click2Mask vs. Emu Edit, as well as Click2Mask vs. MagicBrush, while taking into account "ties" (ratings stating equal performance on an item, or items with equal ratings to both methods), we analyzed the results using the following metrics: (A) The percentage of items in which each method was preferred by the majority, disregarding ties. (B) For each item we counted if the majority voted for a tie, and if so marked it as a "tied item". For the other "non-tied items", we conducted the same majority vote analysis described in A. (C) The number of total ratings for each method. In each parameter our method surpassed the closed-source SoTA method Emu Edit, and the open-sourced SoTA MagicBrush, as shown in Table 1. See Figs. 1, 4 (and 19-22 in appendix) for qualitative comparisons to baselines alongside InstructPix2Pix, and Figure 8 for a detailed comparison. Statistical significance analysis is provided in the appendix.

5.2 Automatic Metrics

Utilizing the input captions and output captions (describing the desired output) provided in Emu Edit benchmark, a variety of metrics were used to assess each method's outputs on the sampled items: (i) Directional CLIP (Gal et al. 2022) similarity (CLIP$_{direct}$) To measure the alignment between changes in input and output images and their corresponding captions. (ii) CLIP similarity between the output image and output caption to evaluate alignment with desired outputs CLIP$_{out}$). (iii) Mean L1 pixel distance between input and output images, to measure the amount of change in the entire image (L1). (iv) In addition, we present a new metric, Edited Alpha-CLIP (αCLIP$_{edit}$).

Edited Alpha-CLIP. Besides evaluating the images *globally*, it is beneficial to evaluate the *edited region*. We offer an Edited Alpha-CLIP procedure to overcome the lack of input or output masks in Emu Edit and MagicBrush: we extract a mask specifying the edited area in the generated image, and calculate the Alpha-CLIP similarity between the masked generated image and the instruction (removing words describing addition and edit locations as mentioned in Section 5). See Appendix A.4 and Figure 12 in appendix for details and extracted masks demonstrations.

Table 2 shows that our method surpassed both Emu Edit and MagicBrush in all metrics: higher scores in all CLIP-based metrics, indicating stronger similarities, and lower L1 distance indicating better compliance with input image.

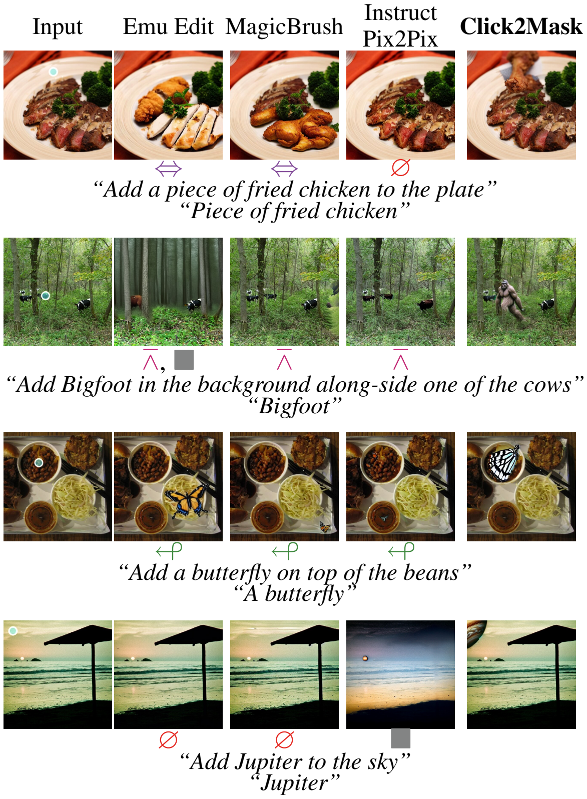


Figure 8: **Failure cases of baselines**. Baselines suffer occasionally from replacing an existing object instead of adding one (⇔), misplacing the object (↫), modifying other objects (⊼), altering the image globally (■), or failing to produce an edit (∅). For additional comparisons to baselines, see Figure 4 and appendix.

5.3 Ablation Study

We conducted several ablation studies to analyze the impact of various components on the overall performance of our model. Figure 7 demonstrates the need for a sufficiently large mask on early diffusion steps. See additional ablation studies in Appendix A.2 accompanied by Figures 13 to 18.

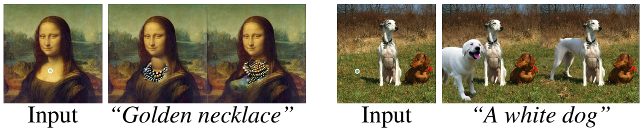


Figure 9: **Limitations.** Top: the evolving mask struggles to converge to a small, fine-detailed shape like a golden necklace. Bottom: when prompt content (i.e. white dog) already exists near the generated mask, our Stable Diffusion and BLD backbones, which guide the image globally to the prompt, may fail to confine guidance to the masked region.

	(A)	(B)		(C)
Method	% Majority	% Tied items	% Majority from non-tied	# Total votes
Emu Edit	42.86%	35%	47.69%	416
Click2Mask	**57.14%**		**52.31%**	**465**
MagicBrush	16.30%	27%	15.07%	148
Click2Mask	**83.70%**		**84.93%**	**362**

Table 1: **Human evaluation results.** Comparisons of (A): % of items each method received majority votes, disregarding ties. (B): % of items the majority voted as tie (left), and % of items – out of the other non-tied items – each method received majority votes (right). (C): Total votes. Refer to Section 5 for details.

Method	CLIP_{direct} ↑	CLIP_{out} ↑	αCLIP_{edit} ↑	L1 ↓
Emu Edit	0.150	0.331	0.186	0.046
MagicBrush	0.095	0.324	0.166	0.049
Click2Mask	**0.204**	**0.334**	**0.195**	**0.027**

Table 2: **Automatic metrics results.** Evaluation using automatic metrics. CLIP_{direct} measures consistency between changes (from input to output) in images and captions, CLIP_{out} measures similarity between output image and output caption, αCLIP_{edit} measures similarity to the non-localized instruction in the edited area, and L1 measures the alignment with the input image. See Section 5 for details.

6 Model Limitations

During the evolution process, our model encounters difficulty converging to small, finely detailed mask shapes (e.g., a dog collar). This stems from hyperparameter choices balancing an initial large mask to capture the object, and a non-aggressive shrinkage rate to avoid boundary cropping. Alternative configurations might achieve smaller masks.

Additionally, since text guidance in Stable Diffusion is not spatially driven, BLD sometimes has difficulty adding the desired object to the masked area when a similar object is nearby in the unmasked area (e.g., adding a Bigfoot next to a person). Since we use BLD as our backbone, we sometimes encounter this problem. However, we have considerably improved it in comparison to BLD by optimizing the progressive mask shrinking process, and applying it across all objects, not just thin objects, as part of our mask evolution process. Moreover, in comparison to other SOTA methods, they often fail to add the desired object even if a similar one is not present, and our method outperforms them in both cases. See Figure 9 for examples of these cases.

7 Conclusion

Click2Mask presents a novel approach for local image generation, freeing users from having to specify a mask, or describing the input or target images, and without being constrained to existing objects. We look forward to users applying our method with the source code that is available in the project page (see Footnote in Page 1), either to edit images or to embed the method for generating or fine-tuning masks.

References

Avrahami, O.; Aberman, K.; Fried, O.; Cohen-Or, D.; and Lischinski, D. 2023. Break-A-Scene: Extracting Multiple Concepts from a Single Image. In *SIGGRAPH Asia 2023 Conference Papers*, SA '23. New York, NY, USA: Association for Computing Machinery. ISBN 9798400703157.

Avrahami, O.; Fried, O.; and Lischinski, D. 2023. Blended Latent Diffusion. *ACM Trans. Graph.*, 42(4).

Avrahami, O.; Lischinski, D.; and Fried, O. 2022. Blended Diffusion for Text-Driven Editing of Natural Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 18208–18218.

Bar-Tal, O.; Ofri-Amar, D.; Fridman, R.; Kasten, Y.; and Dekel, T. 2022. Text2live: Text-driven layered image and video editing. In *European Conference on Computer Vision*, 707–723. Springer.

Betker, J.; Goh, G.; Jing, L.; Brooks, T.; Wang, J.; Li, L.; Ouyang, L.; Zhuang, J.; Lee, J.; Guo, Y.; Manassra, W.; Dhariwal, P.; Chu, C.; Jiao, Y.; and Ramesh, A. 2023. Improving Image Generation with Better Captions.

Brooks, T.; Holynski, A.; and Efros, A. A. 2023. InstructPix2Pix: Learning to Follow Image Editing Instructions. In *CVPR*.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 9650–9660.

Couairon, G.; Verbeek, J.; Schwenk, H.; and Cord, M. 2022. DiffEdit: Diffusion-based semantic image editing with mask guidance. arXiv:2210.11427.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat GANs on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Fu, T.-J.; Hu, W.; Du, X.; Wang, W. Y.; Yang, Y.; and Gan, Z. 2024. Guiding Instruction-based Image Editing via Multimodal Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Gal, R.; Patashnik, O.; Maron, H.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. StyleGAN-NADA: CLIP-guided domain adaptation of image generators. *ACM Trans. Graph.*, 41(4).

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2022. Prompt-to-prompt image editing with cross attention control.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Kawar, B.; Zada, S.; Lang, O.; Tov, O.; Chang, H.; Dekel, T.; Mosseri, I.; and Irani, M. 2023. Imagic: Text-Based Real Image Editing with Diffusion Models. In *Conference on Computer Vision and Pattern Recognition 2023*.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes. arXiv:1312.6114.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; Dollár, P.; and Girshick, R. 2023. Segment Anything. *arXiv:2304.02643*.

Li, F.; Zhang, Z.; Huang, Y.; Liu, J.; Pei, R.; Shao, B.; and Xu, S. 2024. MagicEraser: Erasing Any Objects via Semantics-Aware Control. arXiv:2410.10207.

Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Li, C.; Yang, J.; Su, H.; Zhu, J.; and Zhang, L. 2023. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. arXiv:2303.05499.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2022. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. arXiv:2112.10741.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. arXiv:2204.06125.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E. L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.

Sheynin, S.; Polyak, A.; Singer, U.; Kirstain, Y.; Zohar, A.; Ashual, O.; Parikh, D.; and Taigman, Y. 2023. Emu Edit: Precise Image Editing via Recognition and Generation Tasks. arXiv:2311.10089.

Shi, Y.; Xue, C.; Pan, J.; Zhang, W.; Tan, V. Y.; and Bai, S. 2023. DragDiffusion: Harnessing Diffusion Models for Interactive Point-based Image Editing. *arXiv preprint arXiv:2306.14435*.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Sun, Z.; Fang, Y.; Wu, T.; Zhang, P.; Zang, Y.; Kong, S.; Xiong, Y.; Lin, D.; and Wang, J. 2023. Alpha-CLIP: A CLIP Model Focusing on Wherever You Want. arXiv:2312.03818.

Tumanyan, N.; Geyer, M.; Bagon, S.; and Dekel, T. 2022. Plug-and-Play Diffusion Features for Text-Driven Image-to-Image Translation. arXiv:2211.12572.

Wang, Q.; Zhang, B.; Birsak, M.; and Wonka, P. 2023a. InstructEdit: Improving Automatic Masks for Diffusion-based Image Editing With User Instructions. arXiv:2305.18047.

Wang, S.; Saharia, C.; Montgomery, C.; Pont-Tuset, J.; Noy, S.; Pellegrini, S.; Onoe, Y.; Laszlo, S.; Fleet, D. J.; Soricut, R.; Baldridge, J.; Norouzi, M.; Anderson, P.; and Chan, W.

2023b. Imagen Editor and EditBench: Advancing and Evaluating Text-Guided Image Inpainting. arXiv:2212.06909.

Xie, D.; Wang, R.; Ma, J.; Chen, C.; Lu, H.; Yang, D.; Shi, F.; and Lin, X. 2023. Edit Everything: A Text-Guided Generative System for Images Editing. arXiv:2304.14006.

Xie, S.; Zhang, Z.; Lin, Z.; Hinz, T.; and Zhang, K. 2022. SmartBrush: Text and Shape Guided Object Inpainting with Diffusion Model. arXiv:2212.05034.

Zhang, K.; Mo, L.; Chen, W.; Sun, H.; and Su, Y. 2023. MagicBrush: A Manually Annotated Dataset for Instruction-Guided Image Editing. In *Advances in Neural Information Processing Systems*.

Zou, S.; Tang, J.; Zhou, Y.; He, J.; Zhao, C.; Zhang, R.; Hu, Z.; and Sun, X. 2024. Towards Efficient Diffusion-Based Image Editing with Instant Attention Masks. arXiv:2401.07709.