DM-Align: Text-based semantic image editing using cross-modal alignments

Anonymous ACL submission

Abstract

Text-based semantic image editing assumes the manipulation of an image using a natural language instruction. Although recent works are capable of generating creative and qualitative images, the problem is still mostly approached as a black box sensitive to generating unexpected outputs. Therefore, we propose a novel model to enhance the text-based control of an image editor by explicitly reasoning about which parts of the image to alter or preserve. It relies on word alignments between a description of the original source image and the instruction that reflects the needed updates, and the input image. The proposed Diffusion Masking with word Alignments (DM-Align) allows the editing of an image in a transparent and explainable way. It is evaluated on a subset of the BISON dataset and a self-defined dataset dubbed Dream. When comparing to state-of-the-art baselines, quantitative and qualitative results show that DM-Align has superior performance in image editing conditioned on language instructions, well preserves the background of the image and can better cope with complex text instructions.

1 Introduction

Text-based semantic image editing aims to change the content of a picture by following a text instruction while keeping the remaining visual content untouched. The remaining visual content is from now on referred to as "background". Text-based semantic image editing is usually accomplished using text-based image generation models with user-defined image masks (Avrahami et al., 2022a,b; Wang et al., 2022; Xie et al., 2022). Each of these masks is an arrangement that differentiates between the image content that is to be changed or preserved. However, asking humans to generate masks is cumbersome, so we would like to edit images in a natural way solely relying on a textual description of the image and its instruction to change it. Current models for text-based semantic image editing that do not rely on human-drafted image masks have difficulties in keeping the background (Couairon et al., 2022b; Kwon and Ye, 2022; Couairon et al., 2022a; Choi et al., 2021). Keeping the background static is relevant, especially for crafting games or virtual worlds built by people, where the visual content is expected to be consistent between consecutive frames. Finally, the complexity of the text instructions represents another problem for semantic image editors. While these models can successfully edit images based on short text instructions, they have difficulties in manipulating an image using longer and more elaborate ones.

To tackle the above limitations, we propose a novel method that guides image editing using one-to-one alignments between the words of the text instruction that describes the source image and the textual instruction that describes how the image should look after the editing. Based on word alignments, we can implement an image editing task as a collection of deletion, insertion and replacement operations. Due to text-based control, the proposed model generates good editing results even when the text instructions are long and elaborate, while properly preserving the background.

As presented in Figure 1, we align the words of the text that describes the source image and the textual instruction that describes how the image should look after the editing, which allows us to determine the information the user wants to keep, or replace. Then, disjoint regions associated with the preserved or discarded information are detected by segmenting the image. Next, a global, rough mask for inpainting is generated using standard diffusion models. While the diffusion mask allows the insertion of new objects of different sizes than the replaced ones, it has the disadvantage of being too rough. Therefore, we further refine it using again the detected disjoint regions. To prove the effectiveness of DM-Align, the masked content is generated

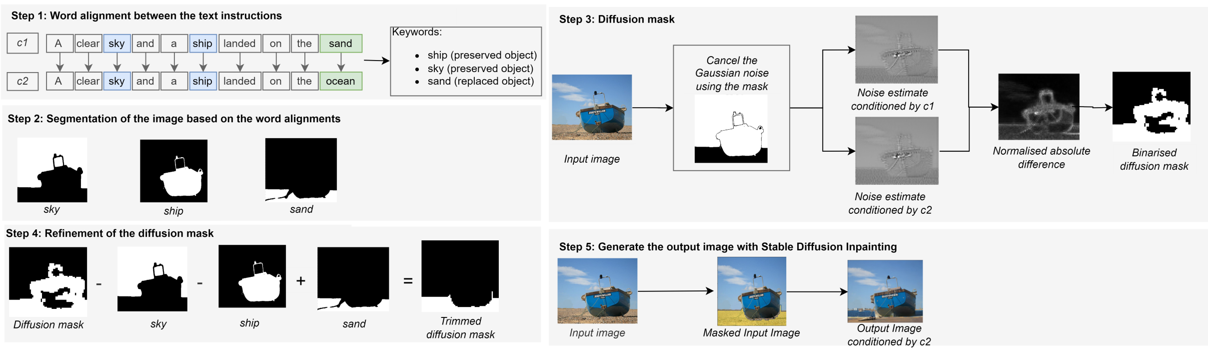

Figure 1: The implementation of DM-Align. The aim is to update the input image described by the text instruction c_1 (“A clear sky and a ship landed on the sand") according to the text instruction c_2 (“A clear sky and a ship landed on the ocean").

using inpainting stable diffusion (Rombach et al., 2022).

Our contributions are summarised as follows:

1. Our novel approach reasons with the text caption of the original input image and the text instruction that guides the changes in the image, which is a natural and human-like way of approaching the problem with a high level of explainability.

2. By differentiating between the image content to be changed from the content to be left unaltered, the proposed DM-Align enhances the text control of semantic image editing.

3. Compared with other recent models designed for text-based semantic image editing, DM-Align can better cope with elaborate and complicated text instructions and can better retain the background of the input image while properly implementing the text instruction.

2 Related work

Despite the aim of keeping the background as similar as possible to the input image, numerous AI-based semantic image editors insert unwanted alterations in the image. FlexIt (Couairon et al., 2022a) combines the input image and instruction text into a single target point in the CLIP multimodal embedding space and iteratively transforms the input image toward this target point. In Kwon and Ye (2022), the image editing is seen as an image translation task that relies on style, and structure losses to guide the training of the model. Zhang and Agrawala (2023) introduce ControlNet as a neural network based on two diffusion models, one frozen and one trainable. While the trainable model is optimized to inject the textual conditionality of the semantic editing, the frozen model preserves the weights of the model pre-trained on large image corpora. The output of ControlNet is gathered by summing the outputs of the two diffusion models. The above approaches lack an explicit delineation of the image content to be altered. Closer to our work is the Prompt-to-Prompt model (Hertz et al., 2022) which connects the text prompt with different image regions using cross-attention maps. The image editing is then performed in the latent representations responsible for the generation of the images. In contrast, our work focuses on the detection and delineation of the content to be altered in the image and is guided by the difference in textual instructions.

To overcome the problem of unwanted alterations in the image, DiffEdit (Couairon et al., 2022b) computes an image mask as the difference between the denoised outputs using the textual instruction that describes the source image and the instruction that describes how the image should look after the editing. However, without an explicit alignment between the two text instructions and the input image, DiffEdit has little control over the regions to be replaced or preserved. While DiffEdit internally creates the editing mask, models like SmartBrush (Xie et al., 2022), Imagen Editor (Wang et al., 2022), Blended Diffusion (Avrahami et al., 2022b) or Blended Latent Diffusion (Avrahami et al., 2022a) directly edit images using hand-crafted user-defined masks.

Due to a rough text-based control, the above models show not only a low ability to preserve the background but also a high sensitivity to the complexity of the text instructions. Different from the current models, our DM-Align model does not treat the recognition of the visual content that requires preservation or substitution as a black box. By explicitly capturing the semantic differences between the natural language instructions, DM-Align is able to comprehensively control the editing of the image, which is novel and leads to better preservation of the image content that needs to remain unaltered and to superior processing of complex text instructions.

3 Proposed model

In this section, we present our solution for semantic image editing. We define the task and then describe the main steps of the proposed model, which consist of 1) detecting the content that needs to be updated or kept relying on the alignment of words of the text that describes the source image and the textual instruction that describes how the image should look after the editing, 2) the segmentation of the image content to be updated or kept by cross-modal grounding, 3) the computation of a global diffusion mask that assures the coherence of the updated image, 4) the refinement of the global diffusion mask with the segmented image content that will be updated or kept and 5) the inpainting of the mask with the help of a diffusion model.

3.1 Task Definition

DM-Align aims to alter a picture described by a source text description or instruction c_1 using a target text instruction c_2. Considering this definition, the purpose is to adjust only the updated content mentioned in the text instruction c_2 and leave the remaining part of the image unchanged. Based on this, we argue the need for a robust masking system that clearly distinguishes between unaltered image regions, which we call “background", and the regions that require adjustments.

3.2 Word alignment between the text instructions

The alignment represents the first step of the DM-Align model proposed to enhance the text-based control for semantic image editing (Figure 1). Given the two text instructions c_1 and c_2, our assumption is that the shared words should indicate unaltered regions, while the substituted words should point to the regions that require manipulations. Implicitly, the most relevant words for this analysis are nouns due to their quality of representing objects in the picture. The words are syntactically classified using the Stanford part-of-speech tagger (Toutanova et al., 2003).

We extend the region to be edited by including the regions of the shared words with different word modifiers[1] in the two text instructions. As a result, the properties of the already existing objects in the picture can be updated. On the contrary, if the aligned nouns have identical modifiers (or no modifiers) in both instructions, their regions in the image should be unaltered. In addition, we also consider the regions of the unaligned nouns mentioned in the source text instruction (deleted nouns) as unaltered regions. Keeping the regions of the deleted nouns is important because we assume that in the target instruction, a user only mentions the desired changes in the image, omitting irrelevant content (Hurley, 2014). Editing the regions of the deleted nouns reduces the similarity w.r.t the source image and increases the level of randomness in the target image since we generate new visual content that is irrelevant to both the source image and the target caption (Figure 7 in Appendix).

The detection of word alignments between the two text instructions is realized with a neural semi-Markov CRF model (Lan et al., 2021). The model is trained to optimize the word span alignments, where the maximum length of spans is equal to D words (in our case $D = 3$). The obtained word span alignments will then further be refined into word alignments.

The neural semi-Markov CRF model is optimized to increase the similarity between the aligned source and target word span representations, which are each computed with a pretrained SpanBERT model (Joshi et al., 2020). The component that optimizes the similarity between these representations is implemented as a feed-forward neural network with Parametric ReLU (He et al., 2015). To avoid alignments that are far apart in the source and target instructions, another component controls the Markov transitions between adjacent alignment labels. To achieve this, it is trained to reduce the distance between the beginning index of the current target span and the end index of the target span aligned to the former source span. Finally, a Hamming distance is used to minimize the distance between the predicted alignment and the gold alignment. The outputs of the above components are

[1] A modifier is a word or phrase that offers information about another word mentioned in the same sentence. To keep the editing process simple, in the current work we use only word modifiers represented by adjectives.

fused in a final function $\psi(a|s,t)$ that computes the score of an alignment a given a source text s and target text t. The conditional probability of span alignment a is then computed as:

$$p(a|s,t) = \frac{e^{\psi(a|s,t)}}{\sum_{a' \in \mathcal{A}} e^{\psi(a'|s,t)}} \quad (1)$$

where the set $\mathcal{A}$ denotes all possible span alignments between source text s and target text t. The model is trained by minimizing the negative log-likelihood of the gold alignment a^* from both directions, that is, source to target $s2t$ and target to source $t2s$:

$$\sum_{s,t,a^*} -\log p(a^*_{s2t}|s,t) - \log p(a^*_{t2s}|t,s) \quad (2)$$

The neural semi-Markov CRF model is trained on the MultiMWA-MTRef monolingual dataset, a subset of the MTReference dataset (Yao, 2014). Considering the trained model, we predict the word alignments as follows. Given two text instructions $c1$ and $c2$, the model predicts two sets of span alignments a: a_{s2t} aligning $c1$ to $c2$; and a_{t2s} aligning $c2$ to $c1$ The final word alignment is computed by merging these two span alignments. Let i be a word of the source text and j be a word of the target text, if alignment a_{s2t} indicates the connection $i- > j$ and alignment a_{t2s} indicates the connection $j- > i$, then the words i and j become aligned. In the end, the word alignments are represented by a set of pairs $(i - j)$, where i is a word of the instruction c_1, and j is a word of the instruction c_2.

3.3 Segmentation of the image based on the word alignments

The aim is to identify the regions in the image that require changes or conservation (second step in Figure 1). Based on the above word alignments, we select the nouns whose regions will be edited (non-identical aligned nouns or aligned nouns with different modifiers in the two text instructions) and the nouns whose regions will stay unaltered (nouns of the source text instruction not shared with the target text instruction, identical aligned nouns). Once these nouns are selected we use Grounded-SAM (Charles, 2023) to detect their corresponding image regions. Its benefit is the "open-set object detection" achieved by the object detector Grounding DINO (Liu et al., 2023) which allows the recognition of each object in an image that is mentioned in the language instruction. Given a noun, Grounding DINO detects its bounding box in the image, and SAM (Kirillov et al., 2023) determines the region of the object inside the bounding box. The selected regions will be used to locally refine the diffusion masks discussed in the next section.

3.4 Diffusion mask

To ensure the coherence of the complete image given the target language instruction and to cope with the different sizes of an object to be replaced and the updated object, we also use a global diffusion mask. To compute the diffusion mask, we first compute the noise estimates of the image corresponding to the source instruction and the noise estimates of the image corresponding to the target instruction by running two separate denoising processes. The noise estimates are obtained using denoising diffusion probabilistic models (DDPM) (Ho et al., 2020). The computation of the diffusion mask represents the third step of our proposed model (Figure 1). The denoising process does not run over the input image but over its encoded representation yielded by a Variational Autoencoder (VAE) (Kingma and Welling, 2014; Rombach et al., 2022) with Kullback-Leibler loss. Therefore, the noise estimates do not represent the final edited image but only an intermediate image representation with semantic information associated with the source or target instruction. By computing the absolute difference between the two noise estimates, we indicate the content to be changed. Meanwhile, the remaining content is irrelevant to the instructions and should stay unaltered. The absolute difference is rescaled between $[0, 1]$ and binarized using a threshold set to 0.5. Details about our implementation with DDPM are presented in Appendix A.

3.5 Refinement of the diffusion mask

The refinement of the diffusion mask represents the fourth step of DM-Align as presented in Figure 1. To further improve the precision of the global diffusion mask, we refine it using the regions detected in Section 3.3. More specifically, we extend the diffusion mask to include the regions to be altered, and shrink it to avoid editing over the preserved regions. To improve control over the preserved background, we adjust the noise variable over the forward process of the obtained diffusion mask. The noise variable is cancelled for the unaltered regions detected in the previous step and kept unchanged for the regions to be manipulated.

Note that both the global diffusion mask with noise cancellation and the regions determined through image segmentation are necessary for a qualitative mask. The global diffusion mask facilitates the replacement of objects of different sizes and gives context to the editing. On the other hand, the insertion or deletion of different regions based on image segmentation improves the precision of the final mask as shown in ablation experiments in Subsection 5.1.

Once the refined diffusion mask is computed, we use inpainting stable diffusion (Rombach et al., 2022) to edit the masked regions based on the given target text caption (fifth step of DM-Align presented in Figure 1). We also tried to replace the inpainting stable diffusion with latent blended diffusion (Avrahami et al., 2022a). However, the obtained results were slightly worse, and the computational time increased by 60% (details are in Table 5 of the Appendix D).

4 Experimental setup

Baselines. We compare results obtained with DM-Align with those of FlexIT (Couairon et al., 2022a), DiffEdit (Couairon et al., 2022b), ControlNet (Zhang and Agrawala, 2023) and Prompt-to-Prompt (Hertz et al., 2022). All results are generated using an NVIDIA Tesla T4 GPU.

Datasets. While the Prompt-to-Prompt paper is missing a quantitative evaluation, FlexIT and DiffEdit are evaluated on a subset of the ImageNet dataset (Deng et al., 2009) that assumes replacing the main object of the scene with another object. Additionally, DiffEdit is evaluated on a subset of the BISON dataset (Hu et al., 2019) and a self-defined collection of Imagen (Saharia et al., 2022) pictures. The quantitative evaluation of ControlNet is limited to only 20 sketches that are not publicly available. Since the datasets that the above works use are not publicly available, we create two datasets, one being a subset of the BISON dataset that we will make publicly available.

Closely following the set-up described in (Couairon et al., 2022b) for creating the subset of the BISON dataset, we use the pairs of similar images and a caption (our source instruction) that describes one of the images in the BISON dataset[2] and obtain the caption of the second image from the COCO 2014 validation dataset (Lin et al., 2014) that functions as a target instruction. Knowing that the BISON dataset is defined for a text-based image classification task and to avoid editing images based on completely unrelated target and source text instructions, a similarity constraint between c_1 and c_2 is imposed. In the current work, we rely on ROUGE-1 (Lin, 2004) to compute the similarity score and set the threshold to 0.7. After applying this filter, we obtain a new dataset with 575 instances. Additional results for different threshold values are discussed in Appendix D (Tables 6-9).

BISON contains complicated and elaborated text captions. To investigate the behaviour of the DM-Align model and the baseline models when confronted with simpler text instructions we generate a collection of 100 images using Dream by WOMBO[3] that relies on the source captions as guidance. To complete the second dataset, we specify a new text query as the target instruction for each image-instruction pair. We further dub the first dataset as $\text{BISON}_{0.7}$ and the second dataset as Dream. When compared with $\text{BISON}_{0.7}$, Dream has a lower complexity with shorter source and target instructions, as one can see in Figure 8, in Appendix. The number of chunks (set of adjacent unigrams in the two instructions aligned by the neural semi-Markov CRF model) observed between the source and target instructions is also smaller in Dream than in $\text{BISON}_{0.7}$ (Figure 8 in Appendix).

Evaluation metrics.

To evaluate our model, we use a set of metrics that assess the similarity of the edited image to both the input image and the target instruction. By default, it is a trade-off between image-based and text-based metrics as we need to find the best equilibrium point.

Generating images close to the source image improves the image-based metrics while reducing the similarity to the target caption. On the other hand, images close to the target instruction improve the text-based scores but can affect the similarity to the input picture. The equilibrium point is important given that people tend to focus mainly on specifying the desired changes in an image while omitting the information that already exists (Hurley, 2014). Therefore, the edited content can represent a small region of the new image while the rest of it should keep the content of the source image.

The similarity (or the distance) of the updated

[2]The BISON dataset was created for the task of associating an image with a descriptive caption

[3]The code is available at https://github.com/cdgco/dream-api

		FID↓	LPIPS↓	PWMSE↓	CLIPScore↑
$\mathrm{BISON}_{0.7}$	FlexIT	72.44 ± 0.15	0.49 ± 0.00	42.34 ± 0.02	$\mathbf{0.88 \pm 0.00}$
	DiffEdit	82.46 ± 0.26	0.46 ± 0.00	50.96 ± 4.07	0.79 ± 0.00
	ControlNet	78.50 ± 0.26	0.42 ± 0.00	52.16 ± 0.78	0.77 ± 0.00
	Prompt-to-Prompt	-	-	-	0.77 ± 0.00
	DM-Align	$\mathbf{60.05 \pm 1.35}$	$\mathbf{0.27 \pm 0.00}$	$\mathbf{34.72 \pm 0.55}$	0.78 ± 0.00
Dream	FlexIT	147.56 ± 1.34	0.71 ± 0.00	53.49 ± 0.01	$\mathbf{0.86 \pm 0.00}$
	DiffEdit	125.71 ± 1.62	0.71 ± 0.00	53.52 ± 0.84	0.77 ± 0.00
	ControlNet	140.18 ± 1.87	$0.72 \pm$	53.78 ± 0.60	0.77 ± 0.00
	Prompt-to-Prompt	-	-	-	0.78 ± 0.00
	DM-Align	$\mathbf{110.20 \pm 0.30}$	$\mathbf{0.69 \pm 0.00}$	$\mathbf{50.62 \pm 0.25}$	0.78 ± 0.00

Table 1: Image-level evaluation for $\mathrm{BISON}_{0.7}$ and Dream datasets (mean and variance). Compared with the baselines, DM-Align achieves the best image-based scores while FlexIT obtains the best similarity w.r.t the target instruction as indicated by CLIPScore. Knowing that the CLIPScore is heavily biased for models based on the CLIP model (as FlexIT does), and considering the image-based scores, DM-Align achieves the best trade-off between similarities to the input image and the target instruction. The image-based metrics of Prompt-to-Prompt are not reported as the method can not edit real images.

		FID↓	LPIPS↓	PWMSE↓
$\mathrm{BISON}_{0.7}$	FlexIT	57.62 ± 0.17	0.22 ± 0.00	21.63 ± 0.00
	DiffEdit	61.23 ± 0.60	0.20 ± 0.00	27.23 ± 2.97
	ControlNet	58.93 ± 0.87	0.19 ± 0.00	18.22 ± 2.02
	DM-Align	$\mathbf{20.17 \pm 1.34}$	$\mathbf{0.05 \pm 0.00}$	$\mathbf{12.24 \pm 0.42}$
Dream	FlexIT	113.06 ± 0.04	0.68 ± 0.00	39.62 ± 0.01
	DiffEdit	72.82 ± 0.14	0.68 ± 0.00	39.34 ± 0.65
	ControlNet	88.23 ± 0.96	0.69 ± 0.00	40.04 ± 0.77
	DM-Align	$\mathbf{41.12 \pm 1.09}$	$\mathbf{0.65 \pm 0.00}$	$\mathbf{36.46 \pm 0.00}$

Table 2: Background-level evaluation for $\mathrm{BISON}_{0.7}$ and Dream datasets (mean and variance). DM-Align outperforms the baselines in terms of background preservation, especially for the dataset $\mathrm{BISON}_{0.7}$ that has more elaborate and complex captions than Dream. The results for Prompt-to-Prompt are not mentioned since the method can not edit real images.

image w.r.t the source image is assessed using FID (Heusel et al., 2017), LPIPS (Zhang et al., 2018) and the pixel-wise Mean Square Error (PWMSE). FID relies on the difference between the distributions of the last layer of the Inception V3 model (Szegedy et al., 2016) that separately runs over the input and edited images. FID measures the consistency and image realism of the new image w.r.t the source image. Contrary to the quality assessment computed by FID, LPIPS measures the perceptual similarity by calculating the distance between layers of an arbitrary neural network that separately runs over the input and updated images. As the LPIPS metric, PWMSE determines the pixel leakage by computing the pixel-wise error between the input and the edited images. The similarity of the updated image w.r.t the target instruction is computed in the CLIP multimodal embedding space by the CLIPScore (Hessel et al., 2021). More details about the evaluation metrics are specified in Appendix B.

5 Results and discussion

5.1 Quantitative analysis and ablation tests

How well can the DM-Align model edit a source image considering the complexity of the text instruction? To answer the first research question, we consider Table 1. Note that Prompt-to-Prompt can not edit real images and therefore, we can only report the CLIPScore. When compared with the baselines Diffedit, ControlNet and FlexIT, the proposed DM-Align model is especially effective w.r.t the image-based metrics. However, this behaviour is more prominent for $\mathrm{BISON}_{0.7}$ that contains elaborate captions. Considering the Dream dataset, DM-Align still scores better than other baselines but with smaller LPIPS and PWMSE margins. However, despite the small margins of FID and LPIPS for the Dream dataset, the difference is still statistically significant w.r.t the best baseline[4].

Both LPIPS and PWMSE rely on mean square error computed either at the level of the internal layers of an arbitrary neural network or at the pixel level. Knowing this, we assume that it is easier for the baselines to correctly edit the image by implicitly creating the correct word alignments between short and simple source and the target instructions. On the contrary, if the text instructions are more elaborate, as in the case of $\mathrm{BISON}_{0.7}$, results are strongly superior compared to those obtained with the baselines. DM-Align relies on word alignments between source and target instructions, showing their importance in effective image editing.

[4]The p-value of the Student's t-test for LPIPS is 0.020 while the p-value for the PWMSE is 0.025. Since the p-values are smaller than the considered significance level equal to 0.05, we reject the null hypothesis and conclude that the difference between DM-Align and the best baseline is statistically significant.

	FID↓	LPIPS↓	PWMSE↓	CLIPScore↑
(w/o) diffusion mask	67.36 ± 1.44	0.33 ± 0.00	34.61 ± 0.26	0.77 ± 0.00
(w/o) noise cancellation	65.30 ± 0.80	0.32 ± 0.00	34.57 ± 0.30	**0.78 ± 0.00**
(w/o) segmentation	76.46 ± 0.20	0.36 ± 0.00	36.47 ± 0.08	0.77 ± 0.00
(w/o) objects with different modifiers	67.53 ± 0.52	0.32 ± 0.00	34.60 ± 0.18	0.77 ± 0.00
(w/o) non-shared objects	68.35 ± 2.25	0.33 ± 0.00	35.34 ± 0.29	0.77 ± 0.00
DM-Align	**60.05 ± 1.35**	**0.27 ± 0.00**	**34.72 ± 0.55**	**0.78 ± 0.00**

Table 3: Ablation tests for the $\mathrm{BISON}_{0.7}$ dataset (mean and variance). The results indicate the importance of the DM-Align components. Non-shared objects refer to the objects mentioned only in the source caption.

With regard to the text-based metrics, the CLIPScore indicates that FlexIT images as the closest to the target instructions. This result is probably explained by the FlexIT architecture which is built on top of a CLIP model which is also used to implement the CLIPScore. This problem is highlighted in (Poole et al., 2022). Another probable explanation is that FlexIT is trained to increase the similarity between the input image and the instructions. As one can see in Figure 2, FlexIT trades off good similarity scores for more distorted images. In terms of CLIPScore DM-Align scores always better than Prompt-to-Prompt and ControlNet, and better than DiffEdit in the case of the Dream dataset.

Overall, DM-Align seems to properly preserve the content of the input image and obtain a better trade-off between closeness to the input picture and target instruction than the baselines. Similar results are observed when comparing DM-Align with baselines using the $\mathrm{BISON}_{0.6}$ and $\mathrm{BISON}_{0.8}$ (Tables 6 and 8 in Appendix D). $\mathrm{BISON}_{0.6}$ represents a subset of BISON obtained by selecting 1437 pairs of source and target captions with ROUGE-1 similarity scores higher than 0.6. $\mathrm{BISON}_{0.8}$ is obtained by setting the ROUGE-1 similarity threshold to 0.8 and counts 105 instances.

How well does the DM-Align model preserve the background? To extract the background, the DM-Align mask obtained after adjusting the diffusion mask is considered. Since Prompt-to-Prompt can not edit real images, this analysis applies only to the other three baselines, DiffEdit, ControlNet and FlexIT. The first thing to observe when analysing results presented in Table 2 is that the FID score of the DM-Align model is reduced by 64.98% for $\mathrm{BISON}_{0.7}$ and by 63.36% for Dream when compared with the best baseline. The LPIPS and PWMSE scores also indicate significant margin reductions, but only for the $\mathrm{BISON}_{0.7}$. These results are similar to the ones observed for the $\mathrm{BISON}_{0.6}$ and $\mathrm{BISON}_{0.8}$ datasets (Tables 7 and 9 in Appendix D).

In the case of the Dream dataset, LPIPS and PWMSE reported for DM-Align are slightly but statistically significant better than the scores of FlexIT, ControlNet and DiffEdit. As observed in Table 1, we infer that the baselines are relatively good at preserving the background only when the instructions are short and simple, but DM-ALign always shows superior results.

Ablation tests According to Table 3, the absence of the refinement of the diffusion mask using the regions detected with the word alignment model and the Grounding-SAM segmentation model has the highest negative impact over the similarity w.r.t the input picture. As expected, a significant negative effect over the similarity with the input image is also noticed when omitting the deleted nouns or the nouns with different modifiers in the two queries. Similarly, noise cancellation and especially the diffusion mask also affect the conservation of the background. Including all the components in the architecture of DM-Align mainly facilitates the preservation of the input image and does not result in a reduction of the CLIPScore. Therefore, the inclusion of all these components in the DM-Align represents the best trade-off w.r.t the similarity to the input image and to the target caption. The ablation tests are exemplified in the Appendix C (Figure 3-7).

5.2 Human qualitative analysis

Some qualitative examples extracted from both data collections are shown in Figure **??**. Since Prompt-to-Prompt does not edit real images, we present its generated images in Figure 9 in Appendix. Without considering the compositional differences due to the unavailability of real images, Prompt-to-Prompt generates less qualitative images when compared with both the other three baselines and DM-Align. Compared to DIFFEdit, ControlNet and FlexIT, the DM-Align model better manipulates the content of the input image and keeps the background w.r.t the target query mostly unchanged. While DM-Align creates semantic con-

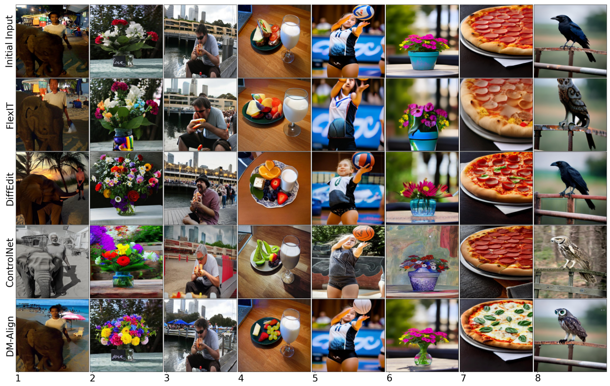

Figure 2: Semantic image editing using $\text{BISON}_{0.7}$ and Dream datasets. **$\text{BISON}_{0.7}$ dataset**: (1) c_2. A man standing next to his elephant on the beach. (2) c_2. A vase filled with lots of colorful flowers. (3) c_2. A man eating a hot dog at a crowded event. (4) c_2. A plate of fruit next to a glass of milk. **Dream dataset**: (5) c_2. A girl throwing a basketball. (6) c_2. A vase with flowers. (7) c_2. A quattro formaggi pizza on a plate. (8) c_1. c_2. An owl sitting on an iron gate.

nections between source and target queries, and updates the image content accordingly, the baselines are limited by the complexity of the text instructions, as discussed above. While DiffEdit changes too much the compositional structure of the image due to the mask-wise correction, FlexIT tends to distort the image. It trades off the minimisation of the reconstruction loss w.r.t. to the input image and the text instructions for possible distortions of the new image. While ControlNet can maintain the structure of the input image, it has difficulties in keeping the texture or colors of the objects. We assume the reason behind the poorer results of ControlNet is the lack of a masking system.

	Q1↑	Q2↑	Q3↑
FlexIt	3.77	4.12	3.83
DiffEdit	3.74	3.89	3.86
ControlNet	3.41	3.77	3.90
Prompt-to-Prompt	2.24	1.98	2.18
DM-Align	**3.89**	**4.35**	**3.95**

Table 4: Human evaluation of the quality of the editing process based on the text instruction (Q1), the preservation of the background (Q2) and the quality of the edited image (Q3). The results represent the average scores reported by annotators using a 5-point Likert scale.

To confirm the above observations, we randomly selected 100 images from the BISON_{07} dataset and asked Amazon MTurk annotators to evaluate the editing quality of the four baselines and the proposed DM-Align. For each edited image, the annotators were asked to evaluate the overall quality of the editing process based on the text instruction (Q1), the preservation of the background (Q2) and the quality of the edited image in terms of compositionality, sharpness, distortion, color and contrast (Q3). According to the human evaluation executed on a 5-point Likert scale, our model scores better than all baselines (Table 4). The inter-rater agreement is good with Cohen's weighted kappa κ between 0.65 and 0.75 for all analysed models.

6 Conclusion, limitations and future work

We propose a novel model DM-Align for semantic image editing that confers to the users a natural control over the image editing by updating the text instructions. By automatically identifying the regions to be kept or altered purely based on the text instructions, the proposed model is not a black box. Due to the high level of explainability, the users can easily understand the edited result and how to change the instructions to obtain the desired output.

The quantitative and qualitative evaluations show the superiority of DM-Align to enhance the text-based control of semantic image editing over existing baselines FlexIT, DiffEdit, ControlNet and Prompt-to-Prompt. Unlike the latter models, our approach is not limited by the complexity of the text instructions. Due to the inclusion of one-to-one alignments between the words of the instructions that describe the image before and after the image update, we can edit images regardless of how complicated and elaborate the text instructions are. Besides the low sensitivity to the complexity of the instructions, the one-to-one word alignments allow us to properly conserve the background while editing only what is strictly required by the users.

DM-Align focuses on the editing of objects mentioned as nouns and their adjectives. In future work, its flexibility can be improved by editing actions in which objects and persons are involved. As a result, they might change position in the image without the need to update their properties.

7 Ethics Statement

Our paper presents a new model for text-based semantic editing without any ethical violation. The data used does not imply any violation of privacy. The potential negative social impacts from this work are similar to any other NLP models.

References

Omri Avrahami, Ohad Fried, and Dani Lischinski. 2022a. Blended latent diffusion. *arXiv preprint arXiv:2206.02779*.

Omri Avrahami, Dani Lischinski, and Ohad Fried. 2022b. Blended diffusion for text-driven editing of natural images. In *2022 Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pages 18187–18197. IEEE.

P.W.D. Charles. 2023. Grounded-sam. https://github.com/IDEA-Research/Grounded-Segment-Anything/tree/main.

Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. 2021. ILVR: conditioning method for denoising diffusion probabilistic models. In *2021 International Conference on Computer Vision (ICCV 2021)*, pages 14347–14356. IEEE.

Guillaume Couairon, Asya Grechka, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2022a. Flexit: Towards flexible semantic image translation. In *2022 Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pages 18270–18279. IEEE.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2022b. DiffEdit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pages 248–255.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 International Conference on Computer Vision*, pages 1026–1034.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2022. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. In *2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, pages 7514–7528. ACL.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *32st Conference on Neural Information Processing Systems (NeurIPS 2017)*, pages 6626–6637.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Annual Conference on Neural Information Processing Systems 2020, (NeurIPS 2020)*.

Hexiang Hu, Ishan Misra, and Laurens van der Maaten. 2019. Evaluating text-to-image matching using binary image selection (BISON). In *2019 International Conference on Computer Vision Workshops (ICCV 2019)*, pages 1887–1890. IEEE.

Patrick J Hurley. 2014. *A concise introduction to logic*. Cengage Learning.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations (ICLR 2014)*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. 2023. Segment anything. *CoRR*, abs/2304.02643.

Gihyun Kwon and Jong Chul Ye. 2022. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*.

Wuwei Lan, Chao Jiang, and Wei Xu. 2021. Neural semi-markov CRF for monolingual word alignment. In *59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP 2021)*, pages 6815–6828. ACL.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. In *13th European Conference on Computer Vision (ECVV 2014)*, volume 8693 of *LNCS*, pages 740–755. Springer.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. 2023. Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. *CoRR*, abs/2303.05499.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *2022 Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pages 10674–10685. IEEE.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pages 2818–2826. IEEE Computer Society.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*. The Association for Computational Linguistics.

Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J. Fleet, Radu Soricut, Jason Baldridge, Mohammad Norouzi, Peter Anderson, and William Chan. 2022. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. *arXiv preprint arXiv:2212.06909*.

Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. 2022. Smartbrush: Text and shape guided object inpainting with diffusion model. *arXiv preprint arXiv:2212.05034*.

Xuchen Yao. 2014. *Feature-driven question answering with natural language alignment*. Ph.D. thesis, Johns Hopkins University.

Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *CoRR*, abs/2302.05543.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 Conference on Computer Vision and Pattern Recognition (CVPR 2018)*, pages 586–595. IEEE Computer Society.

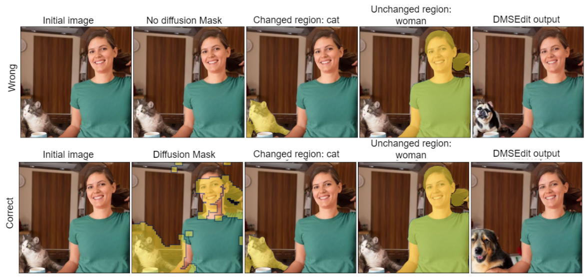

Figure 3: 1st line: Example of omitting the diffusion mask (c_1: A woman near a cat., c_2: A woman near a dog.). 2nd line: The correct example of including the diffusion mask.

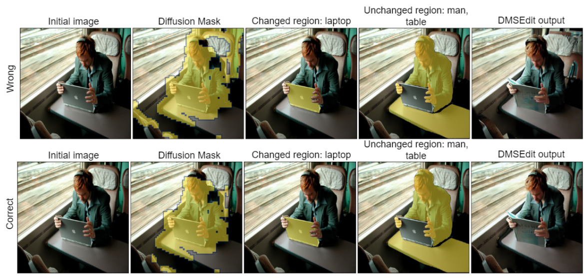

Figure 4: 1st line: Example of omitting the cancellation of the noise variable defined within the diffusion model. (c_1: A man sitting at a table holding a laptop on the train., c_2: A man sitting at a table reading a book on the train.). 2nd line: The correct example of including the noise cancellation.

A Denoising diffusion probabilistic models with noise cancellation

DDPMs are based on Markov chains that gradually convert the input data into Gaussian noise during a forward process, and slowly denoise the sampled data into newly desired data during a reverse process. In each iteration t of the forward process, new data x_t is sampled from the distribution $q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{1-\beta}x_{t-1}, \beta I)$, where β_t is an increasing coefficient that varies between 0 and 1 and controls the level of noise for each time step t. The process is further simplified by expressing the sampled data x_t w.r.t the input data x_0, as follows:

$$x_t = \sqrt{\alpha_t}x_0 + \sqrt{1-\alpha_t}\epsilon \quad (3)$$

where $\alpha_t = \prod_{i=0}^{t}(1-\beta_i)$ and $\epsilon \sim \mathcal{N}(0,1)$ represents the noise variable and is set to 0 over the regions that should be preserved. The process is executed for T iterations until x_T converges to $\mathcal{N}(0,1)$. During the reverse process, at each time step $t-1$, the data is denoised from the distribution $p_\theta(x_{t-1}|x_t) = \mathcal{N}(\sqrt{\alpha_{t-1}}x_0 + \sqrt{1-\alpha_{t-1}-\sigma_t^2}\frac{x_t-\sqrt{\alpha_t}x_0}{\sqrt{1-\alpha_t}})$, where σ^2 represents the variance. After the definition of the two processes, the training of DDPM relies on the varia-

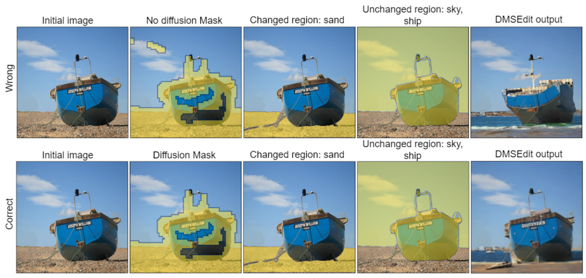

Figure 5: 1st line: Example of omitting the refinement of the diffusion mask using image segmentation (c_1: A clear sky and a ship landed on the sand., c_2: A clear sky and a ship landed on the ocean.). 2nd line: The correct example of including the refinement of the diffusion mask with image segmentation.

Figure 6: 1st line: Example of omitting the information about modifiers associated with the nouns shared by both captions (c_1: A woman with a red jacket., c_2: A woman with a green jacket.). 2nd line: The correct example of including the information about the modifiers.

tional lower bound as follows:

$$log(p(x_0) \geq logp_\theta(x_0|x_1) - D_{KL}(q(x_{1:T}|x_0)||p(x_{1:T}|x_0)) = L_0 - \sum_{t=1}^{T} L_t \tag{4}$$

where D_{KL} represents the Kullback–Leibler divergence, L_0 is the reconstruction loss, L_T shows the proximity of x_T to the Gaussian noise and L_t (t = $\overline{1, T-1}$) indicates the closeness between the denoised step $p(x_t|x_{t+1})$ and the approximated one $q(x_t|x_{t+1})$.

As in the work of Couairon et al. (2022b), the variance of the forward process is set to 0, meaning that we rely on the denoising diffusion implicit models (DDIM), a special case of DDMPs. According to DDIM models, while the forward process becomes deterministic, the model is still trained on the DDPM objective. We use already pre-trained stable diffusers, which means that we are interested to apply DDIM only in terms of sampling. In the current implementation, we run the denoising process of the stable diffusion model for 50 iterations.

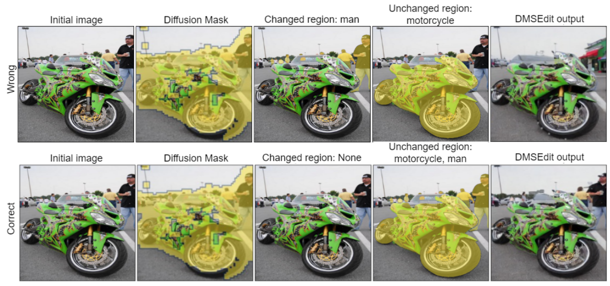

Figure 7: 1st line: Example of omitting the information about the deleted nouns from the source caption (c_1: A motorcycle near a man., c_2: A motorcycle.). 2nd line: The correct example of including the information about the deleted nouns.

B Evaluation Metrics

Image-based evaluation metrics:

- The FID score relies on the distribution of the output generated by the last layer of the Inception V3 model (Szegedy et al. 2016). The metric is computed by measuring the Frechet distance between the distributions gleaned by running the Inception V3 model over the source and target images. Considering the mean μ_1 and the covariance C_1 of the source images and the mean μ_2 and the covariance C_2 of the target images, the FID score is computed as follows:

$$FID = \|\mu_1 - \mu_2\|_2^2 + Tr(C_1 + C_2 - 2(C_1 C_2)^{1/2}) \tag{5}$$

- LPIPS measures the average Euclidean distance between outputs of different layers of a neural network (AlexNet for the current study, as suggested by Zhang et al. (2018)) obtained by giving as input the source and the target images. Considering $x_1^l, \hat{x}_2^l \in \mathcal{R}^{H_l \times W_l \times C_l}$ as the intermediate l-th representations of the AlexNet for the source and the predicted target image, respectively, the LPIPS score is defined by:

$$LPIPS = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \|{x_1}_{hw}^l - (\hat{x}_2)_{hw}^l\|_2^2 \tag{6}$$

- PWMSE measures the pixel-wise mean square error between the input and the edited image.

Text-based evaluation metrics:

- CLIPScore measures the cosine similarity between the CLIP text embedding c_{clip} and CLIP image embedding v_{clip}. The metric is computed as $2.5 * max(cos(c_{clip}, v_{clip}), 0)$. Following the indication of Hessel et al. (2021), CLIP latent embedding space is computed using a Vision Transformer for image encoding and a Transformer for text encoding.

C Visualisations of the Masking Behaviour

The next five visualizations exemplify the ablation tests. The first row of each figure presents the effect of omitting a component of DM-Align, while the correct behaviour is shown in the second row. Figure 3 illustrates the effect of defining the editing mask based only on the image regions of the keywords. Without the diffusion mask, the model has to insert a new object in the fixed area of the replaced object. If we need to replace an object with a larger one, DM-Align without diffusion might create distorted and unnatural outputs. As we usually expect bigger dogs than cats, DM-Align with diffusion properly replaces the cat with a slightly bigger dog. On the contrary, the dog that replaced the cat is distorted when diffusion is not used.

While the overall diffusion mask can give more context for the editing and allows the insertion of objects of different sizes, noise cancellation is an important step used to improve the initial diffusion mask. As shown in Figure 4, when noise cancellation is used, the initial diffusion mask is better trimmed, and the background is properly preserved.

As the diffusion mask does not have complete control over the regions to be edited, its extension or shrinkage based on the image regions of the keywords is mandatory to obtain a correct mask for editing. When the image is edited using only the initial diffusion mask in Figure 5, both the ship and the sand are modified, while the former is expected to be preserved. As opposed, when the diffusion mask is refined with image segmentation, only the sand is replaced by the ocean.

The omission of the adjective modifiers in the analysis of DM-Align is exemplified in Figure 6. If the modifiers are left out, DM-Align considers the jacket a shared noun, like the noun “woman", and removes its regions from the diffusion mask. As a result, DM-Align does not detect any semantical difference between the text instructions, and the output image is identical to the input image. On the other hand, if the modifiers are considered, DM-Align can properly adjust the color of the jacket while keeping the woman’s face unaltered.

As we are interested to make only the necessary updates in the picture, while keeping the background and the regions of the deleted words unchanged, the region assigned to the word “man" in Figure 7 is removed from the diffusion mask. As a result, the corresponding region is untouched. On the contrary, the inclusion of the region associated with the word “man" in the diffusion mask increases the randomness in the new image by inserting a store. Since the store is irrelevant, both the similarity scores w.r.t the input image or target instruction are reduced.

D Additional results

Table 5 presents the results of the comparison between Stable Diffusion and Blended Latent Diffusion for editing the masked regions detected by DM-Align. According to all image-based and text-based metrics, Stable Diffusion confers more robust editing capabilities than Blended Latent Diffusion and it is therefore used to implement DM-Align. Tables 6 and 8 present the image-level evaluation results for $\text{BISON}_{0.6}$ and $\text{BISON}_{0.8}$, while Tables 5 and 7 present the background-level evaluation for the same datasets. Based on the provided results, DM-Aling scores better than all baselines for the image-based metrics while FLexIt still scores better for the CLIPScore due to its architecture.

	FID↓	LPIPS↓	PWMSE↓	CLIPScore↑
DM-Align (Blended Latent Diffusion)	140.87 ± 0.12	0.72 ± 0.00	50.50 ± 0.43	**0.78 ± 0.00**
DM-Align (Statble Latent Diffusion)	**110.20 ± 0.30**	**0.69 ± 0.00**	**50.62 ± 0.25**	**0.78 ± 0.00**

Table 5: Image-level evaluation of DM-Align with Stable diffusion and Blended latent diffusion for inpainting. The results are reported for the Dream dataset (mean and variance).

	FID↓	LPIPS↓	PWMSE↓	CLIPScore↑
FlexIT	41.18 ± 0.07	0.49 ± 0.00	42.51 ± 0.02	**0.89 ± 0.00**
DiffEdit	46.19 ± 0.31	0.47 ± 0.00	50.83 ± 4.14	0.79 ± 0.00
ControlNet	43.67 ± 0.67	0.47 ± 0.00	47.64 ± 2.57	0.78 ± 0.00
Prompt-to-Prompt	-	-	-	0.75 ± 0.00
DM-Align	**33.79 ± 0.12**	**0.28 ± 0.00**	**33.70 ± 0.15**	0.77 ± 0.00

Table 6: Image-level evaluation for $\text{BISON}_{0.6}$ dataset (mean and variance).

	FID↓	LPIPS↓	PWMSE↓
FlexIT	32.30 ± 0.11	0.22 ± 0.00	21.49 ± 0.00
DiffEdit	39.13 ± 0.21	0.22 ± 0.00	24.02 ± 0.18
DiffEdit	34.22 ± 0.53	0.21 ± 0.01	22.02 ± 0.09
DM-Align	**10.28 ± 0.38**	**0.05 ± 0.00**	**12.45 ± 0.22**

Table 7: Background-level evaluation for $\text{BISON}_{0.6}$ dataset (mean and variance).

	FID↓	LPIPS↓	PWMSE↓	CLIPScore↑
FlexIT	112.83 ± 0.08	0.49 ± 0.00	41.61 ± 0.028	**0.88 ± 0.00**
DiffEdit	142.20 ± 0.76	0.46 ± 0.00	51.01 ± 4.07	0.80 ± 0.00
ControlNet	118.56 ± 0.98	0.48 ± 0.00	50.91 ± 2.67	0.81 ± 0.00
Prompt-to-Prompt	-	-	-	0.76 ± 0.00
DM-Align	**96.45 ± 0.34**	**0.27 ± 0.00**	**34.70 ± 0.30**	0.77 ± 0.00

Table 8: Image-level evaluation for $\text{BISON}_{0.8}$ dataset (mean and variance).

	FID↓	LPIPS↓	PWMSE↓
FlexIT	114.86 ± 1.96	0.23 ± 0.00	22.40 ± 0.04
DiffEdit	129.05 ± 1.37	0.21 ± 0.00	28.51 ± 4.17
ControlNet	124.12 ± 1.55	0.21 ± 0.01	22.44 ± 3.98
DM-Align	**34.12 ± 2.09**	**0.05 ± 0.00**	**14.56 ± 0.25**

Table 9: Background-level evaluation for $\text{BISON}_{0.8}$ dataset (mean and variance).

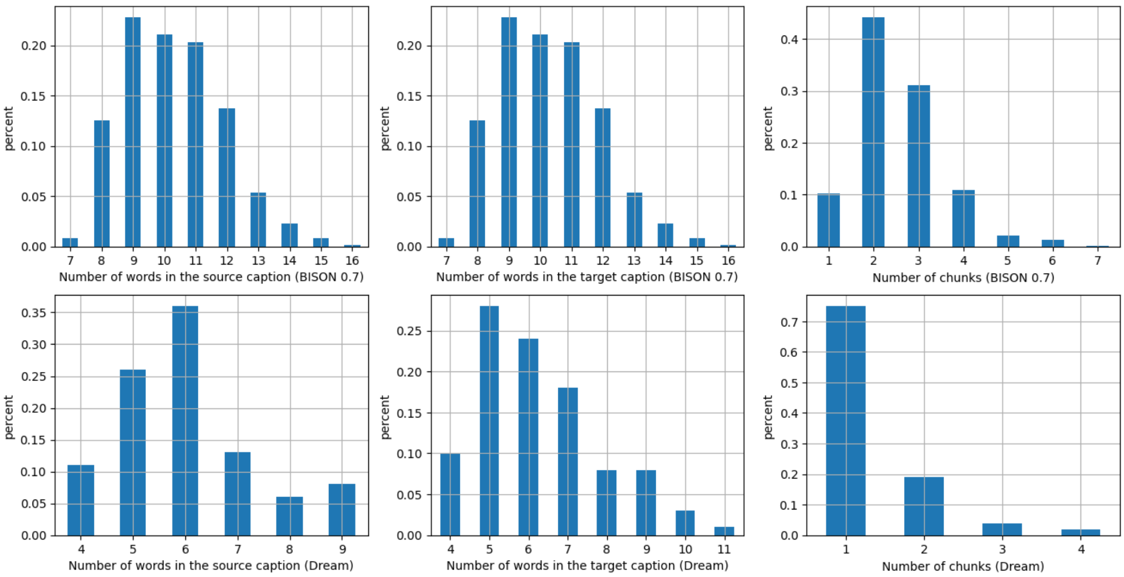

Figure 8: Statistics about $\text{BISON}_{0.7}$ and Dream datasets: number of words in the source and target captions, and number of chunks (set of adjacent unigrams in the two captions aligned by the neural semi-Markov CRF model).

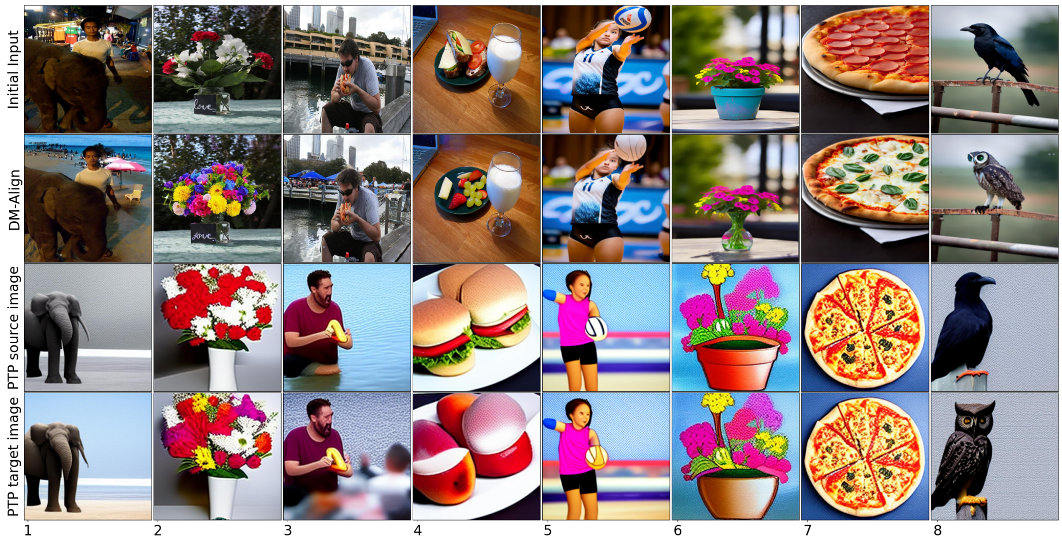

Figure 9: Semantic image editing using $\text{BISON}_{0.7}$ and Dream datasets. **$\text{BISON}_{0.7}$ dataset**: (1) c_1. A man standing next to a baby elephant in the city. c_2. A man standing next to his elephant on the beach. (2) c_1. A vase filled with red and white flowers. c_2. A vase filled with lots of colorful flowers. (3) c_1. A young man eating a hot dog next to a waterway. c_2. A man eating a hot dog at a crowded event. (4) c_1. A plate with open face sandwiches next to a glass of milk and a laptop. c_2. A plate of fruit next to a glass of milk. **Dream dataset**: (5) c_1. A girl throwing a volleyball. c_2. A girl throwing a basketball. (6) c_1. A pot with flowers. c_2. A vase with flowers. (7) c_1. A pepperoni pizza on a plate. c_2. A quattro formaggi pizza on a plate. (8) c_1. A crow sitting on an iron gate. c_2. An owl sitting on an iron gate.