SmartEdit: Exploring Complex Instruction-based Image Editing with Multimodal Large Language Models

Yuzhou Huang[1,2*#] Liangbin Xie[2,3,5*#] Xintao Wang[2,4†] Ziyang Yuan[2,7#] Xiaodong Cun[4]
Yixiao Ge[2,4] Jiantao Zhou[3] Chao Dong[5,6] Rui Huang[1] Ruimao Zhang[1†] Ying Shan[2,4]
[1]The Chinese University of Hong Kong, Shenzhen (CUHK-SZ) [2]**ARC Lab, Tencent PCG** [3]University of Macau [4]Tencent AI Lab
[5]Shenzhen Institute of Advanced Technology [6]Shanghai Artificial Intelligence Laboratory [7]Tsinghua University

https://github.com/TencentARC/SmartEdit

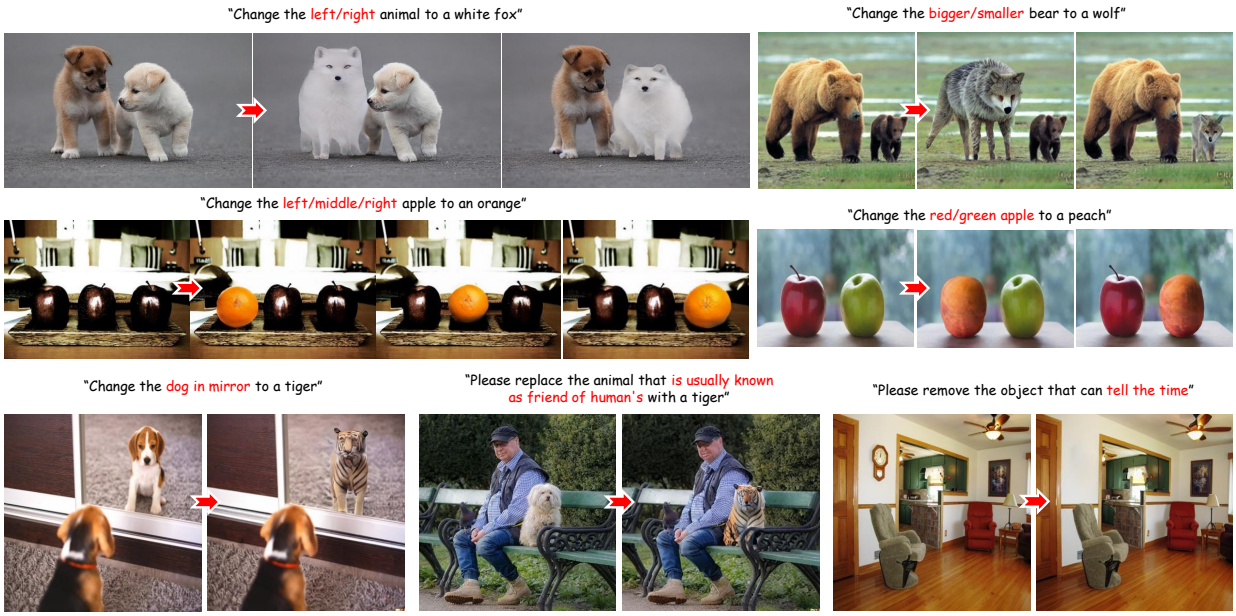

Figure 1. We propose SmartEdit, an instruction-based image editing model that leverages Multimodal Large Language Models (MLLMs) to enhance the understanding and reasoning capabilities of instruction-based editing methods. With the specialized design, our SmartEdit is capable of handling complex understanding (the instructions that contain various object attributes like location, relative size, color, and in or outside the mirror) and reasoning scenarios.

Abstract

Current instruction-based image editing methods, such as InstructPix2Pix, often fail to produce satisfactory results in complex scenarios due to their dependence on the simple CLIP text encoder in diffusion models. To rectify this, this paper introduces SmartEdit, a novel approach of instruction-based image editing that leverages Multimodal Large Language Models (MLLMs) to enhance its understanding and reasoning capabilities. However, direct integration of these elements still faces challenges in situations requiring complex reasoning. To mitigate this, we propose a Bidirectional Interaction Module (BIM) that enables comprehensive bidirectional information interactions between the input image and the MLLM output. During training, we initially incorporate perception data to boost the perception and understanding capabilities of diffusion models. Subsequently, we demonstrate that a small amount of complex instruction editing data can effectively stimulate SmartEdit's editing capabilities for more complex instructions. We further construct a new evaluation dataset, Reason-Edit, specifically tailored for complex instruction-based image editing. Both quantitative and qualitative results on this evaluation dataset indicate that our SmartEdit surpasses previous methods, paving the way for the practical application of complex instruction-based image editing.

* Equal contribution † Corresponding author # Interns in ARC Lab, Tencent PCG

1. Introduction

Text-to-image synthesis [8, 13, 23, 26, 27, 29] has experienced significant advancements in recent years, thanks to the development of diffusion models. These methods have enabled the generation of images that are not only consistent with natural language descriptions, but also align with human perception and preferences, marking a substantial leap forward in the field. Instruction-based image editing methods [1, 36], represented by InstructPix2Pix, leverage pre-trained text-to-image diffusion models as priors. This allows users to conveniently and effortlessly modify images through natural language instructions for ordinary users.

While existing instruction-based image editing methods can handle simple instructions effectively, they often fall short when dealing with complex scenarios, which require the model to have a more powerful understanding and reasoning capabilities. As depicted in Fig. 1, there are two common types of complex scenarios. The first is when the original image contains multiple objects, and the instruction modifies only one of these objects through certain attributes (such as *location, relative size, color, in or outside the mirror*). The other is when world knowledge is needed to identify the object to be edited (such as *the object that can tell the time*). We define these two types as complex understanding scenarios and complex reasoning scenarios, respectively. Handling these two scenarios is crucial for practical instruction editing, but existing instruction-based image editing methods probably fail in these scenarios (as shown in Fig. 2). In this paper, we attempt to identify the reasons why existing instruction-based image editing methods fail in these scenarios, and try to tackle the challenge in these scenarios.

The first reason why existing instruction-based image editing methods fail in these scenarios is that they typically rely on a simple CLIP text encoder [25] in diffusion models (*e.g.*, Stable Diffusion) to process the instructions. Under this circumstance, these models struggle to 1) understand and reason through the instructions, and 2) integrate the image to comprehend the instructions. To address these limitations, we introduce the Multimodal Large Language Models (MLLMs) (*e.g.*, LLaVA) [22, 39] into instruction-based editing models. Our method, SmartEdit, jointly optimizes MLLMs and diffusion models, leveraging the powerful reasoning capabilities of MLLMs to facilitate instruction-based image editing task.

While substituting the CLIP encoder in the diffusion model with MLLMs can alleviate some problems, this approach still falls short when it comes to examples that necessitate complex understanding and reasoning. This is because the input image to edit (original image) is integrated into the UNet of the Stable Diffusion model through a straightforward concatenation, which is further interacted with MLLM outputs through a cross-attention operation. In this setup, the image features serve as the query, and MLLM outputs act as the key and value. This means that the MLLM outputs *unilaterally* modulate and interact with the image feature, which affects the results. To alleviate this issue, we further propose a Bidirectional Interaction Module (BIM). This module reuses the image information extracted by the LLaVA's visual encoder from the input image. It also facilitates a comprehensive bidirectional information interaction between this image and the MLLM output, enabling the model to perform better in complex scenarios.

The second reason contributing to the failure of existing instruction-based editing methods is the absence of specific data. When solely training on editing datasets, such as the datasets used in Instructpix2pix [1] and MagicBrush [36], SmartEdit also struggles to handle scenarios requiring complex reasoning and understanding. This is because SmartEdit has not been exposed to data from these scenarios. One straightforward approach is to generate a substantial amount of paired data similar to those scenarios. However, this method is excessively expensive because the cost of generating data for these scenarios is high.

In this paper, we find that there are two keys to compensate the insufficiency of specific editing data. The first is to enhance the perception capabilities of UNet [28], and the second is to stimulate the model capacity in those scenarios with a few high-quality examples. Correspondingly, we 1) incorporate the perception-related data (*e.g.*, segmentation) into the model's training. 2) synthesize a few high-quality paired data with complex instructions to fine-tune our SmartEdit (similar to LISA [19]). In this way, SmartEdit not only reduces the reliance on paired data under complex scenarios but also effectively stimulates its ability to handle these scenarios.

Equipped with both the model designs and the data utilization strategy, SmartEdit can understand complex instructions, surpassing the scope that previous instruction editing methods can do. To better evaluate the understanding and reasoning ability of instruction-based image editing methods, we collect the Reason-Edit dataset, which contains a total of 219 image-text pairs. Note that there is no overlap between the Reason-Edit dataset and the small-amount high-quality synthesized training data pairs. Based on the Reason-Edit dataset, we evaluate existing instruction-based image editing methods comprehensively. Both the quantitative and qualitative results on the Reason-Edit dataset indicate that SmartEdit significantly outperforms previous instruction-based image editing methods.

In summary, our contributions are as follows:

1. We analyze and focus on the performance of instruction-based image editing methods in more complex instructions. These complex scenarios have often been overlooked and less explored in past research.
2. We leverage MLLMs to better comprehend instructions.

Figure 2. For more complex instructions or scenarios, InstructPix2Pix fails to follow the instructions.

To further improve the performance, we propose a Bidirectional Interaction Module to facilitate the interaction of information between text and image features.

3. We propose a new dataset utilization strategy to enhance the performance of SmartEdit in complex scenarios. In addition to using conventional editing data, we introduce perception-related data to strengthen the perceptual ability of UNet in the diffusion process. Besides, we also add a small amount of synthetic editing data to further stimulate the model's reasoning ability.
4. An evaluation dataset, Reason-Edit, is specifically collected for evaluating the performance of instruction-based image editing tasks in complex scenarios. Both qualitative and quantitative results on Reason-Edit demonstrate the superiority of SmartEdit.

2. Related Work

2.1. Image Editing with Diffusion Models.

Pretrained text-to-image diffusion models [8, 13, 23, 26, 27, 29] can strongly assist image editing task. Instruction-based image editing task [1, 4, 11, 12, 16, 17, 32, 36, 38] requires users to provide an instruction, which converts the original image to a newly designed image that matches the given instruction. Some methods can achieve this by utilizing a tuning-free approach. For example, Prompt-to-Prompt [12] suggests modifying the cross-attention maps by comparing the original input caption with the revised caption. MasaCtrl [4] converts existing self-attention in diffusion models into mutual self-attention, which can help query correlated local contents and textures from source images for consistency. In addition, due to the scarcity of paired image-instruction editing datasets, the pioneering work InstructPix2Pix [1] introduces a large-scale vision-language image editing datasets created by fine-tuned GPT-3 [2] and Prompt-to-Prompt with stable diffusion, and further fine-tunes the UNet [28], which can edit images by providing a simple instruction. To enhance the editing effect of InstructPix2Pix on real images, MagicBrush [36] further provides a large-scale and manually annotated dataset for instruction-guided real image editing.

The recent work, InstructDiffusion [11], also adopts the network design of InstructPix2Pix and focuses on unifying vision tasks in a joint training manner. By taking advantage of multiple different datasets, it can handle a variety of vision tasks, including understanding tasks (such as segmentation and keypoint detection) and generative tasks (such as editing and enhancement). Compared with InstructDiffusion, our primary focus is on the field of instruction-based image editing, especially for complex understanding and reasoning scenarios. In these scenarios, InstructDiffusion typically generates inferior results.

2.2. LLM with Diffusion Models

The exceptional open-sourced LLaMA [6, 31] significantly enhances the performance of vision tasks with the aid of Large Language Models (LLMs). Pioneering works such as LLaVA and MiniGPT-4 have improved image-text alignment through instruction-tuning. While numerous MLLM-based [7, 22, 24, 39] studies have demonstrated their robust capabilities across a variety of tasks, primarily those reliant on text generation (e.g., human-robot interaction, complex reasoning, science question answering, etc.), GILL [18] serves as a bridge between MLLMs and diffusion models. It learns to process images with LLMs and is capable of generating coherent images based on the input texts. SEED [9] presents an innovative image tokenizer to enable LLM to process and generate images and text concurrently. SEED-2 [10] further refines the tokenizer by aligning the generation embedding with the image embedding of unCLIP-SD, which allows for better preservation of rich visual semantics and reconstruction of more realistic images. Emu [30] can be characterized as a multimodal generalist, trained with the next-token-prediction objective. CM3Leon [35] proposes a multi-modal language model that is capable of executing text-to-image and image-to-text generation. It employs the CM3 multi-modal architecture that is fine-tuned on diverse instruction-style data, and utilizes a training method adapted from text-only language models.

3. Preliminary

The goal of instruction-based image editing is to make specific modifications to an input image x based on instructions c_T, resulting in the target image y. InstructPix2Pix, which is based on latent diffusion, is a seminal work in this field. For the target image y and an encoder $\mathcal{E}$, the diffusion process introduces noise to the encoded latent $z = \mathcal{E}(y)$, resulting in a noisy latent z_t, with the noise level increasing over timesteps $t \in T$. A UNet ϵ_δ is then trained to predict the noise added to the noisy latent z_t, given the image condition c_x and text instruction condition c_T, where $c_x = \mathcal{E}(x)$. The

image condition is incorporated by directly concatenating c_x and z_t. The specific objective of latent diffusion is as follows:

$$L_{\text{diffusion}} = \mathbb{E}_{\mathcal{E}(y),\mathcal{E}(x),c_T,\epsilon\sim\mathcal{N}(0,1),t}[\|\epsilon - \epsilon_\delta(t, \text{concat}[z_t, \mathcal{E}(x)], c_T)\|_2^2 \quad (1)$$

where ϵ is the unscaled noise, t is the sampling step, z_t is latent noise at step t, $\mathcal{E}(x)$ is the image condition, and c_T is the text instruction condition. The concat corresponds to the concatenation operation.

Although InstructPix2Pix has some effectiveness in instruction editing, its performance is limited when dealing with complex understanding and reasoning scenarios. To address this issue, we introduce a Multimodal Large Language Model (MLLM) into the network architecture and propose a Bidirectional Interaction Module (BIM) to implement bidirectional information interaction between the MLLM output and image information. In addition, we also explore the data utilization strategy and find that perception-related data and a small amount of complex editing data are crucial for enhancing model's performance. We provide detailed descriptions of these aspects in the next section.

4. Method

In this paper, we introduce SmartEdit, specifically designed to handle complex instruction editing scenarios. In this section, we first provide a detailed overview of the framework of SmartEdit (Section 4.1). Then, we delve into the Bidirectional Interaction Module (Section 4.2). In Section 4.3, we discuss how to enhance the perception and understanding capabilities of UNet in the diffusion model and stimulate the ability of MLLMs to handle complex scenarios. Finally, We introduce Reason-Edit, which is primarily used to evaluate the ability of instruction-based image editing methods toward complex scenarios. (Section 4.4).

4.1. The Framework of SmartEdit

Given an image x and instruction c, which is tokenized as (s_1, ..., s_T), our goal is to obtain the target image y based on c. As shown in Fig 3, the image x is first processed by the image encoder and FC layer, resulting in $v_\mu(x)$. Then $v_\mu(x)$ is sent into the LLM along with the token embedding (s_1, ..., s_T). The output of the LLM is discrete tokens, which cannot be used as the input for subsequent modules. Therefore, we take the hidden states corresponding to these discrete tokens as the input for the following modules. To jointly optimize LLaVA and the diffusion model, following GILL [18], we expand the original LLM vocabulary with r new tokens [IMG$_1$],...,[IMG$_r$] and append the r [IMG] tokens to the end of instruction c. To be specific, we incorporate a trainable matrix $\mathbf{E}$ into the embedding matrix of the LLM, which represents the r [IMG] token embeddings. Subsequently, we minimize the negative log-likelihood of generated r [IMG] tokens, conditioned on tokens that have been generated previously:

$$L_{\text{LLM}}(c) = -\sum_{i=1}^{r} \log p_{\{\theta\cup\mathbf{E}\}}([\text{IMG}_i] \mid v_\mu(x), s_1, ..., s_T, [\text{IMG}_1], \ldots, [\text{IMG}_{i-1}]) \quad (2)$$

The majority of parameters θ in the LLM are kept frozen and we utilize LoRA [15] to carry out efficient fine-tuning. We take the hidden states h corresponding to the r [IMG] tokens as the input for the next module.

Considering the discrepancy between the feature spaces of the hidden states in the LLM and the clip text encoder, we need to align the hidden states h to the clip text encoder space. Inspired by BLIP2 [20] and DETR [5], we adopt the QFormer Q_β with 6−layer transformer [33] and n learnable queries, obtaining feature f. Subsequently, the image feature v output by the image encoder E_ϕ interacts with f through a bidirectional interaction module (BIM), resulting in f' and v'. The mentioned process is represented as:

$$\begin{aligned} h &= \text{LLaVA}(x, c), \\ f &= Q_\beta(h), \\ v &= E_\phi(x), \\ f', v' &= \text{BIM}(f, v) \end{aligned} \quad (3)$$

For the diffusion model, following the design of Instructpix2pix, we concat the encoded image latent $\mathcal{E}(x)$ and noisy latent z_t. Unlike Instructpix2pix, we use f' as the key and value in UNet, and combine v' into the features before entering UNet in a residual manner. The specific process can be formulated as:

$$L_{\text{diffusion}} = \mathbb{E}_{\mathcal{E}(y),\mathcal{E}(x),c_T,\epsilon\sim\mathcal{N}(0,1),t}[\|\epsilon - \epsilon_\delta(t, \text{concat}[z_t, \mathcal{E}(x)] + v', f')\|_2^2 \quad (4)$$

To keep consistency with equation 1, we omit the Conv operation here.

4.2. Bidirectional Interaction Module

The design of BIM is depicted in Fig 4. It includes a self-attention block, two cross-attention blocks, and an MLP layer. The two inputs of BIM are the output f from QFormer and the output v from the image encoder. After bidirectional information interaction between f and v, BIM will eventually output f' and v'. In BIM, the process begins with f undergoing a self-attention mechanism. After this, f serves as a query to interact with the input v, which acts as both key and value, through a cross-attention block. This interaction results in the generation of f' via a point-wise MLP. Following the creation of f', it then serves as both key and value to interact with v, which now acts as a query. This second cross-attention interaction leads to the production of v'.

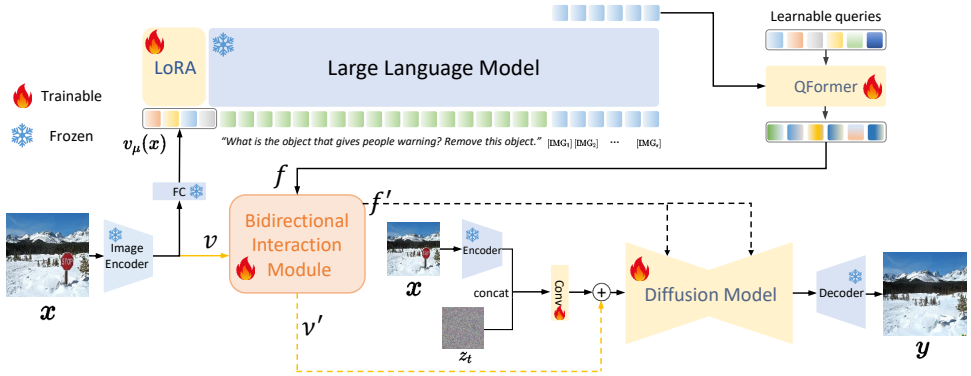

Figure 3. The overall framework of SmartEdit. For the instruction, we first append the r [IMG] tokens to the end of instruction c. Together with image x, they will be sent into LLaVA, which can then obtain the hidden states corresponding to these r [IMG] tokens. Then the hidden state is sent into the QFormer and gets feature f. Subsequently, the image feature v output by the image encoder E_ϕ interacts with f through a bidirectional interaction module (BIM), resulting in f' and v'. The f' and v' are input into the diffusion models to achieve the instruction-based image editing task.

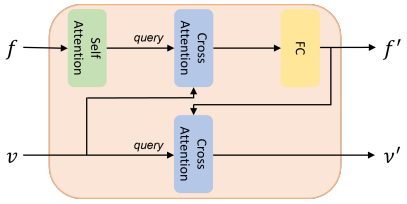

Figure 4. The network design of the BIM Module. In this module, the input information f and v will undergo bidirectional information interaction through different cross-attention.

As discussed in the introduction, the proposed BIM module reuses the image feature and inputs it as supplementary information into UNet. The implementation of two cross-attention blocks in this module facilitates a robust bidirectional information interaction between the image feature and the text feature. Compared to not adopting the BIM module or only fusing the image feature and text feature in one direction, SmartEdit which is equipped with the BIM module yields better results. The experimental comparison of different designs is shown in Section 5.3.

4.3. Dataset Utilization Strategy

During the training process of SmartEdit, two primary challenges emerge when solely utilizing datasets gathered from InstructPix2Pix and MagicBrush as the training set. The first challenge is that SmartEdit has a poor perception of position and concept. The second challenge is that, despite being equipped with MLLM, SmartEdit still has limited capability in scenarios that require reasoning. In summary, the effectiveness of SmartEdit in handling complex scenarios is limited if it is only trained on conventional editing datasets. After analysis, we have identified the causes of these issues. The first issue stems from the UNet in the diffusion model which lacks an understanding of perception and concepts, leading to SmartEdit's poor perception of position and concept. The second issue is that SmartEdit has limited exposure to editing data that requires reasoning abilities, which in turn limits its reasoning capabilities.

To tackle the first issue, we incorporate the segmentation data into the training set. Such modifications significantly enhanced the perception capabilities of the SmartEdit model. Regarding the second issue, we take inspiration from LISA [19] that a minimal amount of reasoning segmentation data can efficiently activate MLLM's reasoning capacity. Guided by this insight, we establish a data production pipeline and synthesize 476 paired data (each sample contains original image, instruction, and synthetic target image) as a supplement to the training data. This synthetic editing dataset includes two major types of scenarios: complex understanding and reasoning scenarios. For complex understanding scenarios, the original image contains multiple objects and the corresponding instruction modifies the specific object based on various attributes (i.e., location, color, relative size, and in or outside the mirror). We specifically consider the mirror attribute because it is a typical example that requires a strong understanding of the scene (both inside and outside the mirror) to perform well. For reasoning scenarios, we involve complex reasoning cases that need world knowledge to identify the specific object. The effectiveness of this synthetic editing dataset and the impact of different datasets on the model's performance are detailed in Section 5.4. The details of the data production pipeline and some visual examples are described in the supplementary material.

4.4. Reason-Edit for Better Evaluation

To better evaluate existing instruction editing methods and SmartEdit's capabilities in complex understanding and reasoning scenarios, we collect an evaluation dataset, Reason-Edit. Reason-Edit consists of 219 image-text pairs. Con-

sistent with the synthetic training data pairs, Reason-Edit is also categorized in the same manner. Note that there is no overlap between the data in Reason-Edit and the synthetic training set. With Reason-Edit, we can thoroughly test the performance of instruction-based image editing models in terms of understanding and reasoning scenarios. We hope more researchers will pay attention to the capabilities of instruction-based image editing models from these perspectives, thereby fostering the practical application of instruction-based image editing methods.

5. Experiments

5.1. Experimental Setting

Training Process and Implementation Details. The training process and implementation details of SmartEdit can be found in the supplementary material.

Network Architecture. For the Large Language Model with visual input (e.g., LLaVA), we choose LLaVA-1.1-7b and LLaVA-1.1-13b as the base model. During training, the weights of LLaVA are frozen and we add LoRA for efficient fine-tuning. In LoRA, the values of the two parameters, dim and alpha, are 16 and 27, respectively. We expand the original LLM vocabulary with 32 new tokens. The QFormer is composed of 6-layer transformer [33] and 77 learnable query tokens. In the BIM module, there is a self-attention block, two cross-attention blocks, and an MLP layer.

Training Datasets. For the training process of SmartEdit, the training data can be divided into 4 categories: (1) segmentation datasets, which include COCOStuff [3], RefCOCO [34], GRefCOCO [21], and the reasoning segmentation dataset from LISA [19]; (2) editing datasets, which involve InstructPix2Pix and MagicBrush; (3) visual question answering (VQA) dataset, which is the LLaVA-Instruct-150k dataset [22]; (4) synthetic editing dataset, where we collect a total of 476 paired data for complex understanding and reasoning scenarios.

Evaluation Metrics. As we hope to only change the foreground of the image while keeping the background unchanged for editing, we adopt three metrics for the background area: PSNR, SSIM, and LPIPS [14, 37]. For the foreground area, we calculate CLIP Score [25] between the foreground area of the edited image and the GT label. The GT label is annotated manually. Among these four metrics, except for LPIPS where lower is better, the other three metrics are higher the better. While these metrics can reflect the performance to a certain extent, they are not entirely accurate. To provide a more accurate evaluation of the effects of edited images, we propose a metric for assessing editing accuracy. Specifically, we hire four workers to manually evaluate the results of these different methods on Reason-Edit. The evaluation criterion is whether the edited image aligns with the instruction. After obtaining the evaluation results from each worker, we average all the results to get the final metric result, which is Instruction-Alignment (Ins-align).

5.2. Comparison with State-of-the-Art Methods

We compare SmartEdit with existing SOTA instruction-based image editing methods, namely InstructPix2Pix, MagicBrush, and InstructDiffusion. Considering that these released models are trained on specific datasets, they would inevitably perform poorly if directly evaluated on Reason-Edit. To ensure a fair comparison, we fine-tune these methods on the same training set used by SmartEdit, and evaluate the fine-tuned models on Reason-Edit. The experimental results are shown in Tab. 1. From the quantitative results of the reasoning scenarios in the table, it can be observed that when we replace the clip text encoder in the diffusion model with LLaVA and adopt the proposed BIM module, both SmartEdit-7B and SmartEdit-13B achieve better results on these five metrics. This suggests that in scenarios requiring reasoning from instructions, a simple clip text encoder may struggle to understand the meaning of the instructions. However, the MLLM can fully utilize its powerful reasoning ability and world knowledge to correctly identify the corresponding objects and perform edits.

The qualitative results further illustrate this point. As shown in Fig. 5, the first three examples are reasoning scenarios. In the first example, both SmartEdit-7B and SmartEdit-13B successfully identify the tool used for cutting fruit (knife) and remove it, while keeping the rest of the background unchanged. The second example can also be handled well by both of them. However, in the third example, we observe a difference in performance. Only SmartEdit-13B can accurately locate the object and perform the corresponding edits without altering other background areas. This suggests that in instruction-based image editing tasks that require reasoning, a more powerful MLLM model can effectively generalize its reasoning ability to this task. This observation aligns with the findings from LISA.

However, for understanding scenarios, we observe a difference in performance between SmartEdit-7B and SmartEdit-13B when compared to InstructDiffusion on PSNR/SSIM/LPIPS. Specifically, SmartEdit-7B performs worse than InstructDiffusion, while SmartEdit-13B outperforms InstructDiffusion on these metrics. Upon further analysis of the qualitative results, as shown in the 4_{th} and 5_{th} rows of Fig. 5, we find that from a visual perspective, both SmartEdit-7B and SmartEdit-13B appear superior to InstructDiffusion. This suggests that the three metrics do not always align with human visual perception. We confirm this phenomenon in the supplementary material. From the result of the Ins-align metric, it can be observed that SmartEdit shows a significant improvement compared to previous instruction-based image editing methods. Also, when adopting a more powerful MLLM model, SmartEdit-13B performs better than SmartEdit-7B on Ins-align.

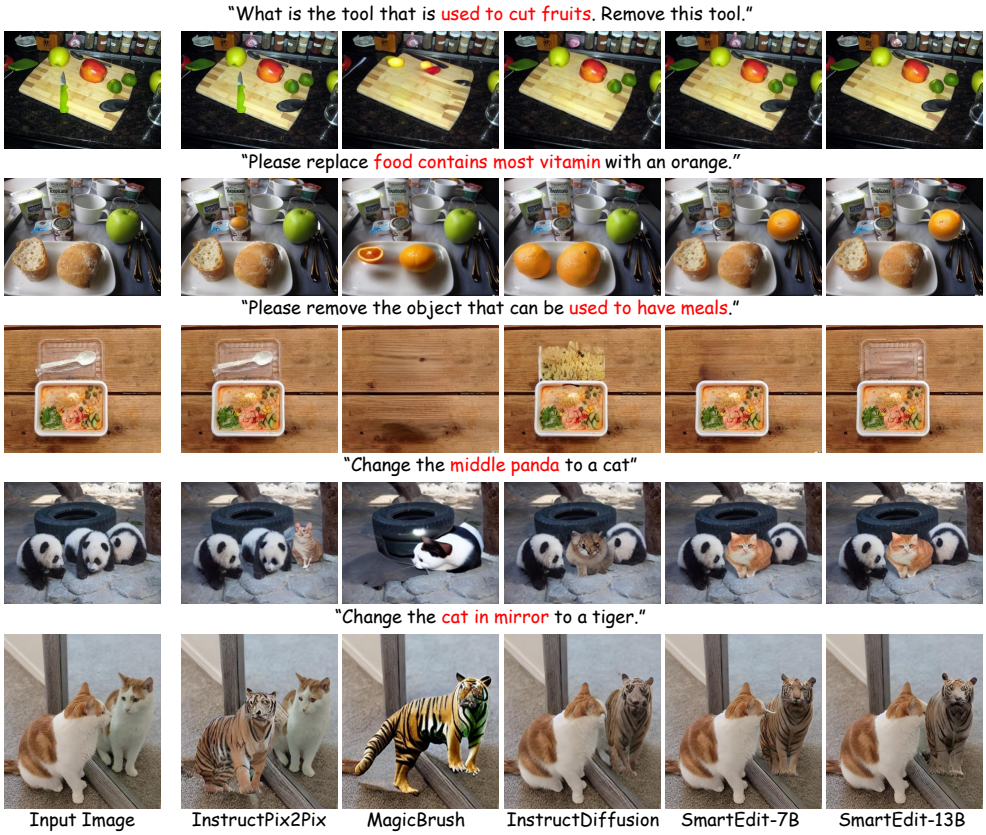

Figure 5. Qualitative comparison on Reason-Edit. When compared to several existing instruction editing methods that have undergone further fine-tuning on our synthetic editing dataset, our approach demonstrates superior editing capabilities in complex scenarios.

Methods	Understanding Scenarios					Reasoning Scenarios				
	PSNR(dB)↑	SSIM↑	LPIPS↓	CLIP Score↑	**Ins-align↑**	PSNR(dB)	SSIM	LPIPS	CLIP Score	**Ins-align↑**
InstructPix2Pix	21.576	0.721	0.089	22.762	0.537	24.234	0.707	0.083	19.413	0.344
MagicBrush	18.120	0.68	0.143	22.620	0.290	22.101	0.694	0.113	19.755	0.283
InstructDiffusion	23.258	0.743	0.067	23.080	0.697	21.453	0.666	0.117	19.523	0.483
SmartEdit-7B	22.049	0.731	0.087	23.611	**0.712**	25.258	0.742	0.055	20.950	**0.789**
SmartEdit-13B	23.596	0.751	0.068	23.536	**0.771**	25.757	0.747	0.051	20.777	**0.817**

Table 1. Quantitative comparison (PSNR↑/SSIM↑/LPIPS↓/CLIP Score↑(ViT-L/14)/Ins-align↑) on Reason-Edit. All the methods we compared have been fine-tuned using the same training data as that used by SmartEdit.

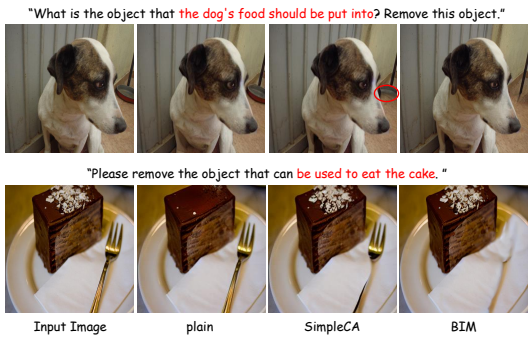

Figure 6. Demonstration of the effectiveness of the BIM Module.

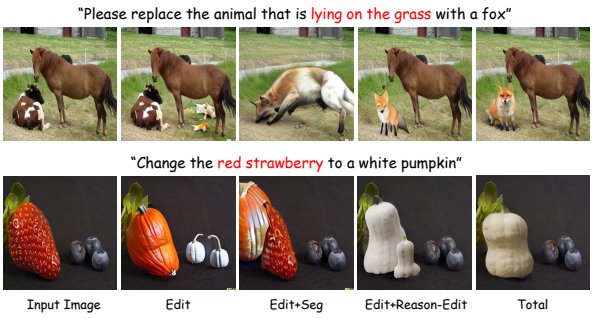

Figure 7. Demonstration of the significance of joint training with multiple datasets.

5.3. Ablation Study on BIM

To validate the effectiveness of the bidirectional information interaction in our proposed BIM module, we conduct comparative experiments on the SmartEdit-7B model. The details are presented in Tab. 2. The first experiment, denoted as Exp 1, aims to verify the necessity of the information in-

Exp ID	Plain	SimpleCA	BIM	Understanding Scenarios					Reasoning Scenarios				
				PSNR(dB)↑	SSIM↑	LPIPS↓	CLIP Score↑	**Ins-align↑**	PSNR(dB)	SSIM	LPIPS	CLIP Score	**Ins-align↑**
1	✓			20.975	0.713	0.108	23.36	0.695	23.848	0.725	0.074	20.33	0.694
2		✓		19.557	0.692	0.126	23.66	0.692	23.508	0.716	0.081	20.17	0.722
3			✓	22.049	0.731	0.087	23.61	**0.712**	25.258	0.742	0.055	20.95	**0.789**

Table 2. Quantitative comparison (PSNR↑/SSIM↑/LPIPS↓/CLIP Score↑(ViT-L/14)/Ins-align↑) on Reason-Edit. These comparative experiments are conducted based on the SmartEdit-7B.

Exp ID	Edit	Segmentation	Synthetic editing dataset	Understanding Scenarios					Reasoning Scenarios				
				PSNR(dB)↑	SSIM↑	LPIPS↓	CLIP Score↑	**Ins-align↑**	PSNR(dB)	SSIM	LPIPS	CLIP Score	**Ins-align↑**
1	✓			17.568	0.664	0.171	22.79	0.201	22.400	0.706	0.102	19.22	0.233
2	✓	✓		18.960	0.690	0.143	22.83	0.361	21.774	0.693	0.116	19.82	0.311
3	✓		✓	19.562	0.702	0.111	22.32	0.440	23.595	0.715	0.079	20.43	0.567
4	✓	✓	✓	22.049	0.731	0.087	23.61	**0.712**	25.258	0.742	0.055	20.95	**0.789**

Table 3. Quantitative comparison (PSNR↑/SSIM↑/LPIPS↓/CLIP Score↑(ViT-L/14)/Ins-align↑) on Reason-Edit. These comparative experiments are conducted based on the SmartEdit-7B.

teraction proposed in the BIM module. In this experiment, we remove the BIM module from the SmartEdit-7B model and directly apply the text feature output from QFormer to the diffusion model. The second experiment, denoted as Exp 2, aims to verify the necessity of the bidirectional information interaction proposed in the BIM module. Specifically, all blocks are discarded except for the cross-attention block on the image feature branch. Therefore, the information from the text feature of QFormer is unidirectionally applied to the image feature. These two experiments are designed to test the impact of removing or altering the BIM module on the performance of SmartEdit-7B in complex understanding and reasoning scenarios. As shown in Tab. 2, if the BIM module is removed, there is a significant decline in all metrics for both understanding and reasoning scenarios. When the BIM module is replaced with the SimpleCA module, we observe a noticeable decline in all metrics, except for the clip score in understanding scenarios. Further comparison of the qualitative results in Fig. 6 confirms that the introduction of the BIM indeed enhances SmartEdit's instruction editing performance. To be specific, when we do not use the BIM module (i.e., plain), the dog bowl (first row) turns into other objects (marked with a red circle), and the fork (second row) does not change at all. After using SimpleCA, it can be found that the dog bowl and fork have been partially removed. When SmartEdit is equipped with BIM, the dog bowl and fork can be well removed.

5.4. Ablation Study on Dataset Usage

In Section 4.3, we explore an efficient strategy for data utilization, aiming to enhance SmartEdit's capabilities in handling complex understanding and reasoning scenarios. During the training process of SmartEdit, we employ the common editing dataset, segmentation dataset, and the synthetic editing dataset. To validate the significance of these different data types in boosting SmartEdit's performance, we conduct a series of ablation studies, as detailed in Tab. 3. These experiments are based on the SmartEdit-7B model. In Exp 1, we train the model using only the editing data. In Exp 2, we incorporate segmentation data into the training process, building upon Exp 1. In Exp 3, we further add the synthetic editing data to the basis established in Exp 1. The quantitative results of these experiments reveal that segmentation data and synthetic editing data play complementary roles in enhancing the model's performance. This is further corroborated by the visual comparison in Fig. 7. For reasoning scenarios, when adopting only the editing dataset or combining the editing dataset and the segmentation dataset, the performance of SmartEdit is inferior. When the synthetic editing data is incorporated into the editing dataset, SmartEdit can accurately locate the specific objects. However, the output of SmartEdit is also mediocre (the generated fox has obvious artifacts, and two pumpkins are generated). When all these datasets are combined as the training set, the results generated by SmartEdit have a further significant improvement in visual effects.

6. Conclusion

In conclusion, this paper presents SmartEdit, a novel approach to instruction-based image editing that enhances understanding and reasoning capabilities by incorporating the LLMs with visual inputs. By introducing the Bidirectional Interaction Module (BIM), we have overcome challenges associated with the direct integration of LLMs and diffusion models in complex reasoning scenarios. Our data utilization strategy, which incorporates perception data and complex instruction editing data, effectively enhances SmartEdit's capabilities in handling complex understanding and reasoning scenarios. Evaluation on our newly constructed dataset, Reason-Edit, shows that SmartEdit outperforms previous methods, marking a significant step towards practical applications of complex instruction-based image editing.

Acknowledgement. The work is partially supported by the Young Scientists Fund of the National Natural Science Foundation of China under grant No. 62106154, the Natural Science Foundation of Guangdong Province, China (General Program) under grant No.2022A1515011524, Shenzhen Science and Technology Program JCYJ20220818103001002 and JCYJ20220818103006012, and the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong (Shenzhen).

References

[1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 2, 3

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 6

[4] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. *arXiv preprint arXiv:2304.08465*, 2023. 3

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 4

[6] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. 3

[7] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2, 3

[9] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041*, 2023. 3

[10] Yuying Ge, Sijie Zhao, Ziyun Zeng, Yixiao Ge, Chen Li, Xintao Wang, and Ying Shan. Making llama see and draw with seed tokenizer. *arXiv preprint arXiv:2310.01218*, 2023. 3

[11] Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Han Hu, Dong Chen, et al. Instructdiffusion: A generalist modeling interface for vision tasks. *arXiv preprint arXiv:2309.03895*, 2023. 3

[12] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 3

[14] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE, 2010. 6

[15] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 4

[16] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code. *arXiv preprint arXiv:2304.04269*, 2023. 3

[17] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 3

[18] Jing Yu Koh, Daniel Fried, and Ruslan Salakhutdinov. Generating images with multimodal language models. *arXiv preprint arXiv:2305.17216*, 2023. 3, 4

[19] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. *arXiv preprint arXiv:2308.00692*, 2023. 2, 5, 6

[20] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 4

[21] Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23592–23601, 2023. 6

[22] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2, 3, 6

[23] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2, 3

[24] Yiran Qin, Enshen Zhou, Qichang Liu, Zhenfei Yin, Lu Sheng, Ruimao Zhang, Yu Qiao, and Jing Shao. Mp5: A multi-modal open-ended embodied system in minecraft via active perception. *arXiv preprint arXiv:2312.07472*, 2023. 3

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 6

[26] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2, 3

[27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 2, 3

[29] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 2, 3

[30] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*, 2023. 3

[31] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[32] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 3

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4, 6

[34] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016. 6

[35] Lili Yu, Bowen Shi, Ramakanth Pasunuru, Benjamin Muller, Olga Golovneva, Tianlu Wang, Arun Babu, Binh Tang, Brian Karrer, Shelly Sheynin, et al. Scaling autoregressive multimodal models: Pretraining and instruction tuning. *arXiv preprint arXiv:2309.02591*, 2023. 3

[36] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *arXiv preprint arXiv:2306.10012*, 2023. 2, 3

[37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[38] Enshen Zhou, Yiran Qin, Zhenfei Yin, Yuzhou Huang, Ruimao Zhang, Lu Sheng, Yu Qiao, and Jing Shao. Minedreamer: Learning to follow instructions via chain-of-imagination for simulated-world control. *arXiv preprint arXiv:2403.12037*, 2024. 3

[39] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2, 3