Efficient Sketch-Guided Image Inpainting via Composite Condition and Matched RoPE

Anonymous AI Frontline submission

Paper ID

Abstract

We build a complete sketch-guided image inpainting pipeline as a course project. Starting from a strong diffusion transformer baseline (FLUX.1), we systematically compare multiple ways to inject sketch control and finally adopt a parameter-efficient strategy inspired by OminiControl: we (i) construct a single composite conditional image by fusing sketch, mask, and visible pixels, avoiding an explicit mask branch; (ii) append conditional tokens to the visual token sequence; and (iii) align conditional and noisy tokens via specialized rotary positional embeddings (RoPE) so that corresponding spatial locations share identical positional phases. We fine-tune the backbone with LoRA and show that our method maintains high fidelity in the unmasked region while improving structural adherence in the inpainted region, with reduced token length and modest training cost. We also develop a paired sketch–mask–image dataset with targeted augmentations, including free-form deformation (FFD) to mimic user-drawn sketch distortions.

1. Introduction

Sketch-guided inpainting employs a model to fill a missing region of an image such that the generated content is (i) semantically consistent with a text prompt, (ii) structurally consistent with a user-provided sketch, and (iii) seamlessly blended with the visible context. Compared with text-only inpainting, sketch control introduces two practical challenges: (1) *abstraction and distortion*: user sketches are often sparse, non-photorealistic, and spatially misaligned; and (2) *efficient conditioning*: strong structure control typically requires extra networks (e.g., ControlNet-like branches) or additional input tokens, which increases compute and latency.

In this project, we aim to build a robust yet efficient system for sketch-guided inpainting. Our design starts from FLUX.1, a rectified-flow diffusion transformer operating in VAE latent space with a mixture of double-stream and single-stream blocks and factorized 3D RoPE [1]. Inspired by OminiControl-style parameter-efficient control, we focus on *how* to inject sketch conditions into FLUX with minimal additional cost.

Key idea. Instead of feeding sketch and mask as separate modalities (or adding an explicit mask input branch), we build a *single composite condition image* that merges (visible) context pixels, the sketch drawn within the masked region, and the mask itself (encoded implicitly by where we place the sketch vs. original pixels). This composite condition is tokenized and appended to the visual token sequence. Crucially, we align the RoPE indices so that each conditional token and its corresponding noisy token share the same position ID, enabling direct spatial correspondence in attention. This matched RoPE alignment is the core mechanism for strong structure control without learning an additional coordinate mapping.

Contributions.

- A complete sketch-guided inpainting pipeline built on FLUX.1 with parameter-efficient training (LoRA).
- An embedded condition injection strategy: composite condition image + token concatenation + matched RoPE alignment.
- A paired sketch–mask–image dataset construction pipeline with targeted augmentations (including FFD) for robustness to sketch abstraction and misalignment.
- Empirical evaluation showing improved sketch adherence while preserving unmasked-region fidelity with reduced token length.

2. Related Work

Inpainting and diffusion-based editing. Diffusion models have become a dominant paradigm for image generation and editing. Classic inpainting pipelines often inject a binary mask explicitly, either as an extra channel or via specialized U-Net conditioning. Representative diffusion-based inpainting/editing works include RePaint [5] and

large-mask inpainting systems such as LaMa [9] (for non-diffusion baselines), as well as image-to-image editing approaches like SDEdit [6]. These methods highlight the trade-off between control strength and computational overhead.

Sketch-guided generation. Sketch/line-conditioned editing and inpainting have been explored by injecting sparse structural cues (e.g., coarse sketch lines, contours, or wireframe-like representations) into conditional generative models. DeFLOCNet [4] studies flexible low-level controls for deep image editing, and explicitly demonstrates an inpainting setting where coarse sketch lines guide the completion of missing regions. For structure-critical scenes, Cao *et al*. [2] learn a sketch-tensor space that captures lines, edges, and junctions to improve inpainting of man-made environments. In practice, such sketch-like guidance may range from clean edges to abstract strokes, so robustness to drawing style and spatial imprecision remains crucial.

Condition injection frameworks. ControlNet [10] and T2I-Adapter [7] popularize adding an auxiliary control branch to a frozen backbone to steer generation with edges, sketches, depth, pose, etc. While effective, extra branches increase memory and compute. OminiControl (project) emphasizes parameter-efficient control with minimal modifications, motivating our design to reuse the backbone and limit trainable parameters (via LoRA) while preserving strong spatial control.

Diffusion Transformers and rectified flow. DiT [8] reformulates diffusion with transformers on latent tokens. FLUX.1 is a rectified-flow transformer trained in the latent space of an autoencoder, with mixed double-stream and single-stream blocks and factorized 3D RoPE [1]. These architectural choices influence where and how we should inject conditional tokens.

Parameter-efficient fine-tuning. LoRA [3] is widely adopted to adapt large models by learning low-rank updates to attention/MLP layers, reducing training cost. We use LoRA to fine-tune FLUX.1 for sketch-guided inpainting.

3. Preliminaries

3.1. FLUX.1 diffusion transformer

FLUX.1 is a rectified-flow transformer trained in the latent space of an image autoencoder [1]. It mixes *double-stream* blocks (separate weights for image and text streams, with attention applied over concatenated tokens) and *single-stream* blocks (a unified sequence of image and text tokens) [1]. Positional information is encoded with factorized 3D RoPE, where each latent token is indexed by its space-time coordinates (t, h, w) (with $t \equiv 0$ for single images) [1]. This RoPE design is a natural handle for spatially aligned conditioning.

3.2. Sketch-guided inpainting formulation

Given an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$, a binary mask $\mathbf{m} \in \{0,1\}^{H\times W}$ (with $\mathbf{m} = 1$ indicating the *missing* region), a sketch image $\mathbf{s} \in \mathbb{R}^{H\times W\times 3}$ (or single-channel), and a text prompt c, the goal is to synthesize a completed image $\hat{\mathbf{x}}$ such that:

- $\hat{\mathbf{x}}$ matches $\mathbf{x}$ on the visible region $(1 - \mathbf{m})$,
- $\hat{\mathbf{x}}$ respects the structure in $\mathbf{s}$ inside $\mathbf{m}$, and
- the overall output is semantically consistent with c.

3.3. Condition injection perspective

In transformer-based diffusion, image latents are tokenized into a visual sequence. A condition can be injected by: (i) concatenating condition tokens to the visual sequence; (ii) adding cross-attention from visual tokens to condition tokens; or (iii) introducing a separate control branch. We focus on (i) because it is simple and efficient, and it matches FLUX-style "sequence concatenation" conditioning used in FLUX.1 Kontext [1].

4. Method

4.1. Overview

Our method consists of (1) composite condition image construction, (2) embedded condition injection by token concatenation, and (3) matched RoPE alignment. Figure 1 illustrates the pipeline.

4.2. Composite condition image

Motivation. A naive approach feeds the mask explicitly (extra channel or extra tokens) and feeds the sketch as another modality. However, explicit mask inputs increase token length and complicate architecture changes. We instead encode the mask implicitly through a *composite condition image* that uses different content in masked vs. unmasked regions.

Construction. In our formulation, the model is conditioned only on the composite image $\mathbf{I}_{\text{comp}}$; therefore, an explicit binary mask is *optional*. We first present the standard full-resolution construction with an explicit mask, and then describe a more flexible "implicit-mask" variant enabled by our conditioning interface.

(A) Explicit-mask composition. Let $\mathbf{m} \in \{0,1\}^{H\times W}$ be the binary mask, where $\mathbf{m} = 1$ indicates missing pixels. We define

$$\mathbf{I}_{\text{comp}} = (1 - \mathbf{m}) \odot \mathbf{x} + \mathbf{m} \odot \phi(\mathbf{s}), \qquad (1)$$

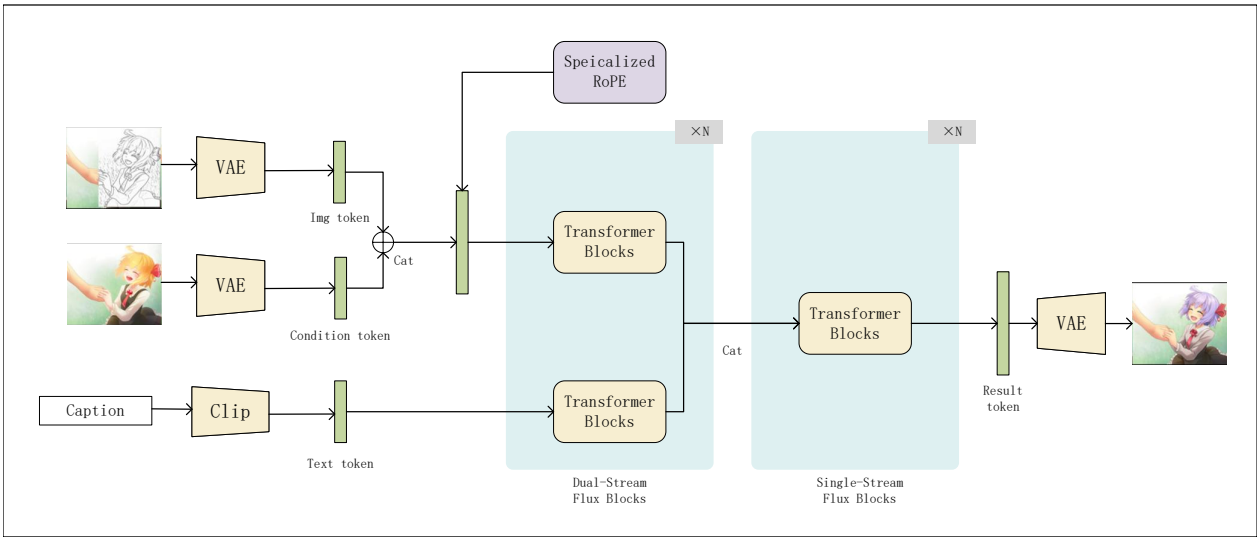

Figure 1. Method overview. We form a composite condition image by fusing the visible pixels and the sketch within the masked region, tokenize it into conditional tokens, append them to the noisy visual tokens, and apply matched RoPE alignment so that conditional and noisy tokens at the same spatial location share identical position IDs.

where $\odot$ is element-wise multiplication and $\phi(\cdot)$ converts the sketch into a raster image aligned to the image canvas (e.g., replicating to 3 channels and normalizing intensities). Intuitively, the model sees ground-truth context in the unmasked region and sees sketch strokes in the masked region.

(B) Implicit-mask composition via sketch placement. Since the backbone only consumes $\mathbf{I}_{\text{comp}}$, we can construct it without providing $\mathbf{m}$ explicitly. Let $\mathbf{s}_0 \in \mathbb{R}^{h\times w(\times 1)}$ be a (possibly smaller) sketch patch and let $\mathcal{P}(\cdot; p)$ denote a placement operator that pastes the sketch patch onto a blank $H \times W$ canvas at location p (e.g., top-left corner at (u, v)), producing an aligned sketch canvas $\tilde{\mathbf{s}} = \mathcal{P}(\mathbf{s}_0; p)$. We then define an *implicit mask* by the support of the placed sketch:

$$\tilde{\mathbf{m}}(i, j) = \alpha(i, j), \tag{2}$$

where $\alpha(i, j) \in \{0, 1\}$ denotes the alpha channel of the placed sketch canvas ($\alpha = 1$ for pixels covered by the pasted sketch strokes, and $\alpha = 0$ for transparent background), and then compose

$$\mathbf{I}_{\text{comp}} = (1 - \tilde{\mathbf{m}}) \odot \mathbf{x} + \tilde{\mathbf{m}} \odot \phi(\tilde{\mathbf{s}}). \tag{3}$$

In this view, "where the sketch is placed" implicitly specifies the region to be edited, making the explicit mask input unnecessary when users provide localized sketches.

Why this helps. This design encourages the model to (i) preserve the unmasked region by directly providing it as condition, and (ii) interpret strokes as a structural hint for missing content. Compared with explicit-mask designs, we reduce the need for a separate mask-processing pathway and can reduce effective input length in some injection variants (see Sec. 4.6): when an explicit-mask baseline injects *both* a sketch condition sequence and a separate mask sequence (each tokenized to N latent tokens), the additional tokens scale as $N_{\text{extra}} = N_{\text{sketch}} + N_{\text{mask}} = 2N$, whereas our composite condition uses a *single* condition sequence ($N_{\text{extra}} = N$), yielding a 50% reduction in extra condition tokens.

4.3. Embedded condition injection via token concatenation

Following FLUX-style conditioning, we encode $\mathbf{I}_{\text{comp}}$ into VAE latents and tokenize it into a conditional token sequence $\mathbf{T}_{\text{cond}}$. Let $\mathbf{T}_{\text{noisy}}$ denote the noisy latent tokens at diffusion time t. We form a visual sequence by concatenation:

$$\mathbf{S}_{\text{visual}} = \text{Concat}(\mathbf{T}_{\text{noisy}}, \mathbf{T}_{\text{cond}}), \tag{4}$$

and then proceed with the FLUX blocks. In FLUX.1, double-stream blocks treat text and image streams separately and interact via attention mechanisms, while single-stream blocks process a unified sequence [1]. We adapt the injection to both stages:

- **Double-stream stage:** $\mathbf{T}_{\text{cond}}$ participates in the image stream attention early, helping establish the global structure topology.
- **Single-stream stage:** $\mathbf{T}_{\text{cond}}$, $\mathbf{T}_{\text{noisy}}$, and text tokens attend jointly, enabling deep fusion of structure (sketch) and semantics (prompt).

4.4. Matched RoPE alignment

Key mechanism. We align positional indices between conditional and noisy tokens. Concretely, instead of assigning new position IDs to conditional tokens, we force them to reuse the position IDs of their spatially corresponding noisy tokens. This makes the RoPE phase identical for a pair of tokens at the same spatial location, enabling attention to directly couple "current pixel" and "corresponding sketch" without learning an additional spatial transform.

Implementation sketch. Let a token at spatial location (h, w) have a RoPE index $\pi(h, w)$. For each (h, w), we set:

$$\pi_{\text{cond}}(h, w) := \pi_{\text{noisy}}(h, w). \tag{5}$$

4.5. Training objective and LoRA fine-tuning

We keep the original rectified-flow / flow-matching objective of the FLUX backbone and fine-tune with LoRA. Let $\mathbf{z}_t$ be a noised latent at time t and $\mathbf{v}_\theta(\cdot)$ be the predicted velocity/flow field. We optimize a flow matching loss:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,\epsilon}\left[\|\mathbf{v}_\theta(\mathbf{z}_t, t; c, \mathbf{I}_{\text{comp}}) - \mathbf{v}^\star\|_2^2\right], \tag{6}$$

where $\mathbf{v}^\star$ is the target velocity under the chosen rectified-flow parameterization.

We apply LoRA adapters to attention projections. This allows fast fine-tuning while keeping most backbone weights frozen.

4.6. Efficiency discussion

Our design is efficient in three senses:

- **No extra control branch:** unlike ControlNet-style add-on networks, we reuse the backbone and only introduce LoRA parameters.
- **Simple conditioning interface:** we use token concatenation, consistent with FLUX-style sequence construction [1].
- **Reduced explicit inputs:** the mask is not injected as a separate explicit modality; instead it is encoded implicitly via the composite condition image, which can reduce extra tokens/channels needed by some baselines.

5. Dataset Preparation

5.1. Paired sketch–mask–image construction

We construct training triplets $(\mathbf{x}, \mathbf{m}, \mathbf{s})$ with the following goals: (1) diverse object categories and complex scenes, (2) sketches that resemble human drawings rather than perfect edges, and (3) masks that simulate realistic user edits.

Hybrid training sources. We firstly try to adapt existing datasets to our tasks, like FSCOCO and SketchyScene, but then we find these datasets are originally intended for drawing from scratch, which has already been achieved by previous works. So we trained them on these datasets to preemptively evaluate the capability of OminiControl Training Framework, which has satisfying results Figure 1.

But when it comes to fitting the 2 datasets for our specific task, problems arises as the sketches are either too roughly drawn by hand or semantically irrelevant as generated by AI so that we failed to extract the main objects from them, for which reason wo cannot build the sketch-implanted images we needed for our specific tasks. Therefore, we shift our minds and discover anime images are always a couple of characters that are distinctively standing out against the background, so we choose the Dabooru Datasets as our training dataset source.

As the dataset is crawled from Dabooru, a original amine image board, the images are of varying quality. So we apply Image Quality Assessment (IQA) metrics to filter the images on technical distortions (like blur, noise, and compression artifacts), structural integrity, and aesthetic appeal.

Object-centric masking and sketch acquisition. To enable semantic-level alignment training with the filtered Danbooru dataset, we leverage a segmentation model (SAM3) in preprocessing. Given an image, the SAM3 proposes object-level regions (in our case, heads/characters) and helps associate sketches with bounding boxes of the highest confidence. We then apply segmentation to extract object masks and define the inpainting region $\mathbf{m}$ in an object-centric manner. As a results, the head/character region on the original image is replaced with the sketch region $\tilde{\mathbf{m}}$.

5.2. Sketch robustness augmentations

Real user sketches are often misaligned and distorted. To improve robustness, we apply a single targeted augmentation during training: **non-rigid free-form deformation (FFD)**. Specifically, we warp sketch strokes using a coarse control lattice with smooth interpolation, which mimics hand-drawn proportion errors and local shape drift while preserving the overall topology. This augmentation discourages the model from memorizing pixel-precise sketch coordinates and encourages more semantic use of structural cues.

6. Experimental Setup

6.1. Baselines

We mainly compare 2 ways to inject sketch guidance into FLUX-like backbones:

- **Text-only inpainting:** We evaluate FLUX under a mask-conditional setting without sketch constraints. Our empirical findings indicate two primary failure modes: (1)

Figure 2. Qualitative results on danbooru (top) and fscoco (bottom). Each example is organized as *input image* | *sketch/condition* | *ours*. Our method faithfully restores the missing regions for small masks while preserving global semantics and structure for full-mask generation, producing coherent textures and natural transitions along mask boundaries.

prompt-alignment degradation, where the model fails to strictly follow complex or long-context text instructions; and (2) descriptive ambiguity, where the intended filling region involves irregular shapes or textures that exceed the expressive capacity of natural language prompts.

- **Ours:** To address the baseline limitations, we employ a composite conditioning strategy that integrates structural guidance (sketches) via image implantation. Crucially, we implement Matched RoPE Alignment: instead of appending sequential positions, we assign the condition tokens the exact same rotary positional indices as the target latents. This enforces strict spatial correspondence between the structural condition and the generated output.

Table 1. Quantitative results on danbooru dataset.

Method	PSNR ↑	SSIM ↑	FID ↓
Ours	15.40	0.78	76.20

6.2. Training details

We fine-tune FLUX.1 with LoRA. Unless otherwise specified, we use: optimizer Prodigy, learning rate 1.0 (with Prodigy) weight decay 0.01, LoRA rank 16, resolution 512*512, and 5,000 training steps for head-sketch implantation and 10,000 training steps for character-sketch implantation, with a flow matching (rectified flow) loss.

6.3. Evaluation metrics

We report PSNR and SSIM to measure pixel-level fidelity and structural similarity, and we also report FID to reflect perceptual realism and distributional similarity between generated outputs and real images. All metrics are computed on the final inpainted images over the evaluation set.

7. Results

7.1. Qualitative results

Figure 2 presents representative results on Danbooru (partial masks) and FSCOCO (full masks). Across examples, our method better follows sparse and abstract strokes inside the masked region while keeping the visible context nearly unchanged. We also observe cleaner transitions along mask boundaries, suggesting that the composite condition helps preserve global consistency and prevents unintended edits outside the target region. Notably, even when sketches contain crossings or mild spatial distortion, matched RoPE alignment encourages location-wise correspondence between conditional and noisy tokens, yielding more plausible structure completion.Figure 3

7.2. Quantitative results

We report PSNR/SSIM to quantify fidelity on the visible (unmasked) region, and FID to reflect perceptual realism of the final outputs. Table 1 summarizes the results on danbooru dataset. Since our project focuses on an efficient condition-injection design and is trained at a course-project scale, we include these metrics as a reference for overall quality; a more exhaustive quantitative comparison against strong sketch-control baselines is left for future work.

(a) Metrics for Original Images

(b) Metrics for Generated Images

Figure 3. Comparative analysis of IQA scores between original and generated datasets.

7.3. Ablation studies

We conduct ablations to isolate the effect of each component:

- **Without composite condition:** feed sketch alone (masked region not distinguished), leading to weaker preservation of context and occasional artifacts.
- **Without matched RoPE:** assign independent position IDs to condition tokens; we observe degraded structural control, especially under sketch misalignment.
- **Without FFD augmentation:** the model overfits to "clean" sketch geometry and becomes brittle to user-like distortions.

Figure 4. **Failure case due to misalignment.** The figure demonstrates the model's behavior when the sketch implantation is not perfectly aligned with the underlying image. The spatial discrepancy between the provided sketch (middle) and the original structure leads to distortion in the final generation (right).

8. Discussion

8.1. Why matched RoPE alignment works

Matched RoPE alignment makes spatial correspondence explicit at the positional-encoding level. In attention, tokens at the same spatial location share the same RoPE phase, encouraging the model to learn a direct mapping between the current latent state and the intended structure implied by the sketch. This is particularly useful when sketches are sparse: the model can propagate structural constraints via global attention while still grounding them spatially.

8.2. Failure cases and limitations

Our model can still fail when sketches are extremely symbolic (e.g., stick figures) or when the sketch implantation is not perfectly aligned with the image Figure 4. We also observe that overly long or repeated edits (multi-turn) can accumulate artifacts (a general limitation in interactive editing models).

8.3. Practical notes for a course project

We prioritize a clean, reproducible pipeline and parameter-efficient training over exhaustive scaling. This choice keeps engineering complexity manageable while still enabling meaningful exploration of condition injection designs.

9. Conclusion

We presented a sketch-guided inpainting system built on FLUX.1 with an efficient control mechanism inspired by OminiControl. Our key design—composite condition image, token concatenation, and matched RoPE alignment—provides strong structural control while preserving unmasked-region fidelity, with modest training

overhead using LoRA. We also described a dataset construction and augmentation pipeline (including FFD) to improve robustness to sketch abstraction and distortion. Future work includes controllable "roughness" (faithfulness-to-sketch) and extending to multi-condition control (e.g., depth/pose) in a unified interface.

References

[1] Black Forest Labs. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 1, 2, 3, 4

[2] Chen Cao and (CVF). Learning a sketch tensor space for image inpainting of man-made scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 2

[3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 2

[4] Hongyu Liu, Ziyu Wan, Wei Huang, Yibing Song, Xintong Han, Jing Liao, Bing Jiang, and Wei Liu. Deflocnet: Deep image editing via flexible low-level controls. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[5] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[6] Chenlin Meng, Yutong He, Yang Song, Stefano Ermon, Jiajun Wu, and Serge Belongie. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2

[7] Chong Mou, Xintao Zhang, Zhaoyang Li, and others. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024. 2

[8] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4172–4182, 2023. 2

[9] Roman Suvorov and others. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022. 2

[10] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2