Exploring Latent Space Manipulation through Point Dragging

赵泽宇 2400017711@stu.pku.edu.cn
王唐欣宇 元培学院 2400017808@stu.pku.edu.cn
鄢宇阳 2400017766@stu.pku.edu.cn

2026.01.03

1 Introduction

1.1 Project Overview

Manipulation of image details is an important topic of research due to its demands in abilities of flexible and precise control of image details. Existing methods such as traditional warping methods or GANs based on manually annotated data/pretrained 3D models often lacks either abilities.

In this project, we will dive into the latent space of images and examine how DragGAN edits the images through "dragging" points in its latent space. The project will not only focus on the principles of the DragGAN model, but also explore pipelines of possible improvements for overcoming some limitations mentioned in the articles.

1.2 Plan

The project will focus on following parts:

- Analysis of the pipeline and code base
- Reproduction of qualitative results(as well as some quantitative results)
- Improvement of model performance on textureless areas and masking performance
- Comparison with DragDiffusion on out-of-distribution manipulation.

2 Problem Statement & Technical Approach

To gain full understanding of the pipeline, it is required to reproduce results and analyse the mixed loss function in the paper. Moreover, the original method exhibits several limitations that can be further improved, which will also be dealt with in this project.

2.1 Improving Tracking Robustness in Texture-less Regions

DragGAN performs image manipulation by modifying the latent vector in the GAN latent space. As a result, it requires an explicit tracking mechanism to locate the same handle points after each optimization iteration. DragGAN adopts a point-wise matching strategy to estimate point displacement across iterations. Although the original paper reports that learning-based trackers such as RAFT are less effective in this setting, point-wise matching becomes highly ambiguous in texture-less regions, which can significantly degrade the robustness of the point tracking pipeline. Global constraints or "motion" information may be helpful in solving this problem.

2.2 Improving Masking Performance and Background Preservation

DragGAN employs a mixed loss function:

$$\mathcal{L} = \sum_{i=0}^{n} \sum_{\boldsymbol{q}_i \in \Omega_1(\boldsymbol{p}_i, r_1)} \|\boldsymbol{F}(\boldsymbol{q}_i) - \boldsymbol{F}(\boldsymbol{q}_i + \boldsymbol{d}_i)\|_1 + \lambda \|(\boldsymbol{F} - \boldsymbol{F}_0) \cdot (1 - \boldsymbol{M})\|_1 , \tag{1}$$

where the second term enforces feature consistency in regions outside the mask, encouraging the background (i.e., pixels with $\boldsymbol{M} = 0$) to remain unchanged.

However, due to the entangled nature of the GAN latent space, strictly preserving the background is inherently challenging. Local manipulations in latent space often induce unintended global changes, as latent directions corresponding to different semantic regions may overlap and influence each other. As a result, subtle background variations are difficult to completely avoid even with explicit masking constraints.

To address this issue, we will try to explore an optimized manipulation pipeline that further disentangles foreground edits from background content, aiming to improve background preservation while maintaining interactive control fidelity.

Moreover, the adoption of simply a global masked L1 loss relies on partial continuity of images and also the assumption that shifted image would have greater pixel-wise difference. However, this does not apply to sharply changing areas where a small shift or a large shift have similar loss, for that the loss would not be able to constrain the scale of shift. Adoption of better shifting penalty(loss) of masked area would also be a possible improvement.

2.3 Comparative Study of DragGAN and DragDiffusion

In the discussion section of DragGAN, the authors point out that the method may suffer from noticeable distortions when manipulating images toward out-of-distribution poses that were rarely or never observed during GAN training. This limitation stems from the intrinsic constraint of GAN-based generative models, whose image manifold is bounded by the training data distribution.

To better understand this limitation and its practical implications, we will conduct a comparative study between DragGAN and DragDiffusion.

3 Progress and Intermediary Results

During the milestone stage, we have thoroughly analyzed the paper and successfully run the official code base. We gained a clear understanding of the overall pipeline and implementation details, and conducted preliminary reproductions of the qualitative results. In addition, to better understand the proposed approach, we reviewed the related work referenced in the paper, including GAN-based methods and StyleGAN. The following section presents our analysis of the DragGAN pipeline.

3.1 Motivation from StyleGAN

A key observation of StyleGAN is that higher layers control fine-grained details, while lower layers govern the spatial structure and content. DragGAN, on the other hand, is largely independent of the classic GAN framework regarding the interaction between the generator and discriminator; it primarily utilizes the generator. In DragGAN, the point-wise distance loss (Equation 1) is applied at the sixth layer, and the gradient is backpropagated to the $\boldsymbol{w}$ vectors of the first six layers, which are subsequently optimized to move the control points closer to their target locations.

3.2 Analysis of DragGAN with Code

Based on our analysis of `viz/renderer.py`, the DragGAN optimization involves an iterative loop of point tracking and motion supervision.

3.2.1 Point Tracking

Handle points are tracked via feature matching (Lines 328-341). For each point, the algorithm searches for the nearest neighbor in the feature map within a local patch (Line 336) by minimizing the L2 distance (Line 337) to update its location.

3.2.2 Loss Function

The optimization is driven by a composite loss function in `_render_drag_impl`:

- **Motion Loss** (Line 360): L1 distance between current features and target features, driving points towards targets.
- **Fixed Area Loss** (Line 366): L1 difference ensuring unmasked regions remain unchanged.
- **Regularization Loss** (Line 369): L1 regularization on the latent vector $\boldsymbol{w}$ to preserve image quality.

3.2.3 Optimization

Gradients are backpropagated to the latent code $\boldsymbol{w}$ (Line 372). The optimizer updates $\boldsymbol{w}$ to minimize the accumulated loss from all points simultaneously.

3.3 Some Reproduction Results

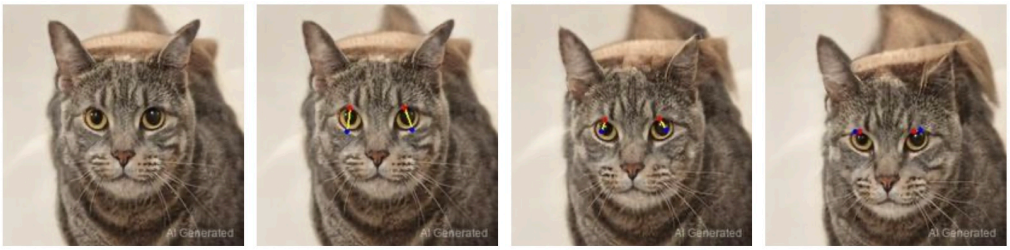

Figure 1: Reproduction result on an AI-generated cat image. The intended manipulation was to close the cat's eyes; however, when only drag points were applied, the transformation predominantly affected the position of the cat's head instead.

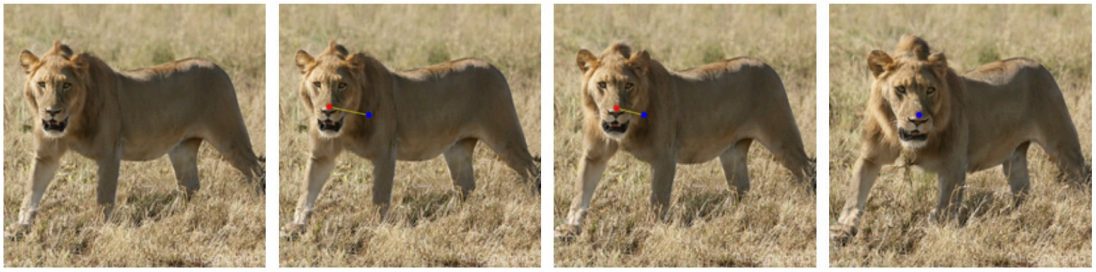

Figure 2: Reproduction result on an AI-generated lion image with the goal of translating the lion's head. When using drag points alone, the transformation caused the lion's body to move simultaneously with the head, indicating limited motion disentanglement.

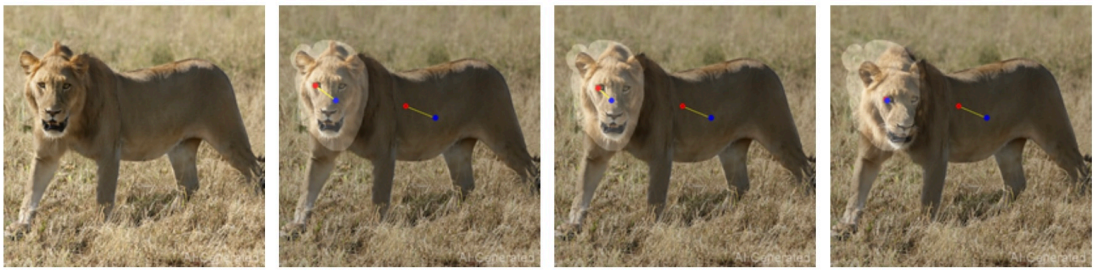

Figure 3: Reproduction result on an AI-generated lion image aiming to translate the lion's head. With the combined use of drag points and a spatial mask, the head was successfully manipulated independently from the body. Additionally, fine-grained facial characteristics of the female lion are better preserved compared to the previous result.

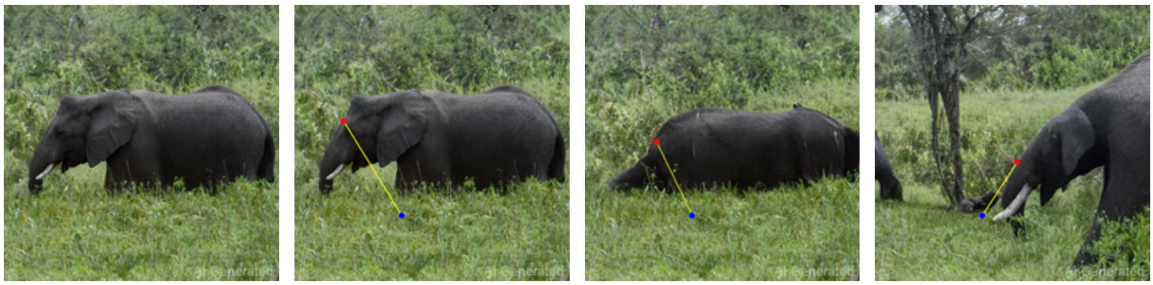

Figure 4: Reproduction result on an AI-generated elephant image with a large-magnitude head translation. During the intermediate stage, the elephant's head features were significantly degraded due to excessive deformation, but were unexpectedly recovered in the final output. Nevertheless, noticeable and unintended alterations in the background were observed.

Figure 5: Reproduction results on an AI-generated horse image involving coordinated leg translation. The first group presents results obtained using a four-point dragging strategy, while the second group uses an eight-point dragging strategy. More precise and stable control is achieved when explicit control points are assigned to all ankle joints.

Figure 6: Reproduction result on an AI-generated car image under a large-magnitude pose transformation. The vehicle is successfully rotated as intended, while preserving its overall shape and scale without noticeable geometric distortion.

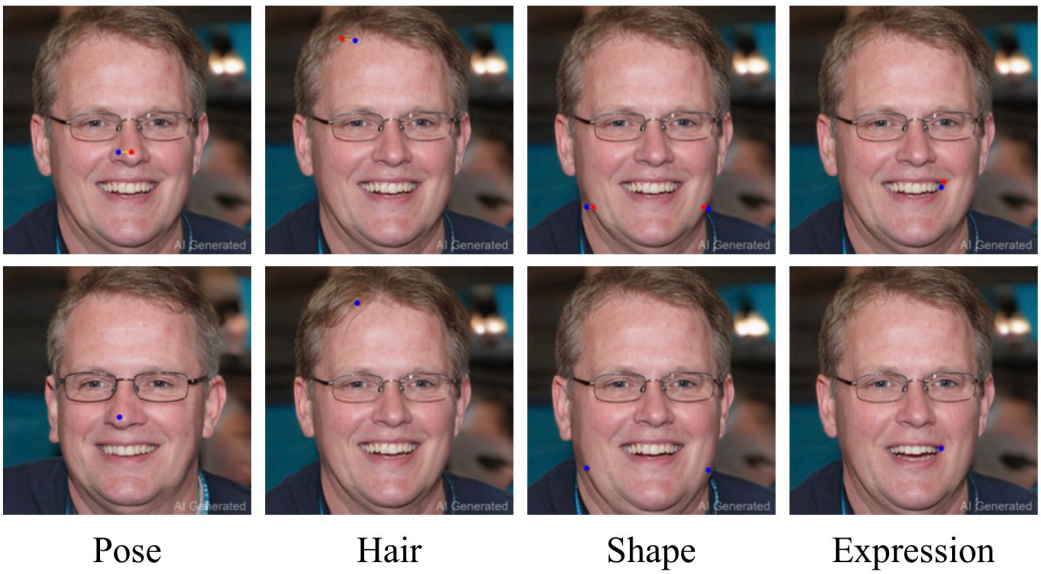

Figure 7: Reproduction results on an AI-generated human image demonstrating multiple fine-grained editing tasks, including (1) pose adjustment, (2) hairstyle modification, (3) body shape refinement, and (4) facial expression manipulation. The results indicate that DragGAN effectively preserves the structural consistency of the main subject and background during localized edits, and performs particularly well on in-domain tasks such as pose and expression changes.