Controllabilities in Video Diffusion Models

Junoh Kang

Computer Vision Laboratory
ECE, Seoul National University
junoh.kang@snu.ac.kr

May 10th, 2024

Computer Vision Lab
Seoul National University

Contents

How can we describe video?

To describe a typical video, we have to explain

- Appearance of objects,
- Motions of objects,
- Background, the place where objects are in,
- *e.t.c.* ...

There are more but let us think about only three aspects!

How can we describe video?

Texts

First, I willl use only **text**, the most convinient mean for human.

- Appearance of objects : A girl wearing a white dress.
- Motions of objects : Dancing is not enough. Crossing hands where the second and fifth finger spreaded, then shaking hands up and down, ...
- Background: Purple lighted room with mirror floor.

How can we describe video?

Texts

Texts are not enough to describe video!

- Appearance of objects : It does not specify the girl.
- Motions of objects : We cannot dance with only text descriptions above.
- Background : May be this is enough for now, but not for complicated one.

How can we describe video?

Images

Then, let me explain the three of the following video with **images**.

How can we describe video?

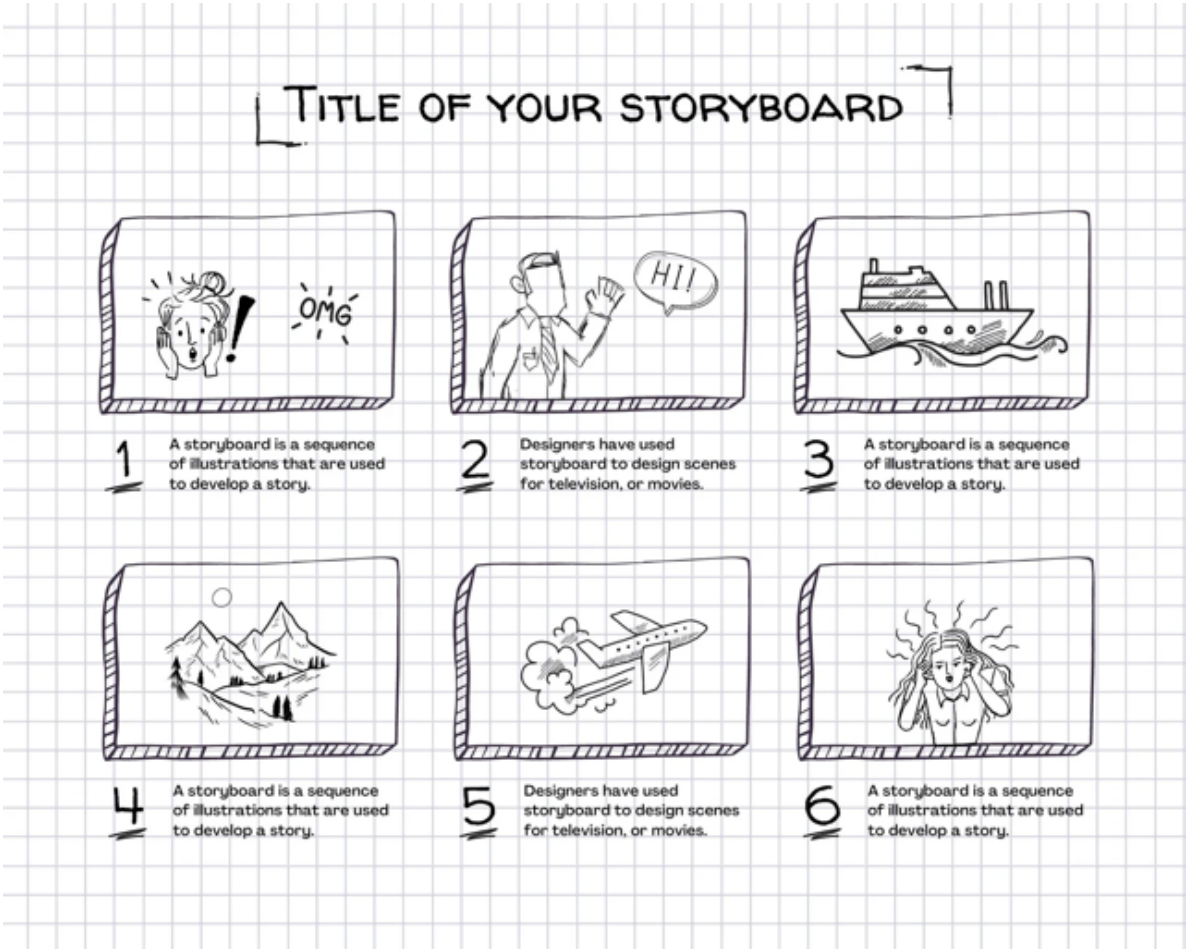
Images

- Appearance of objects : Pretty good description.
- Motions of objects : We need fine snapshots to explain the motion and we do not know the intermediate actions.
- Background : Pretty good description.

How can we describe video?

Texts and Images

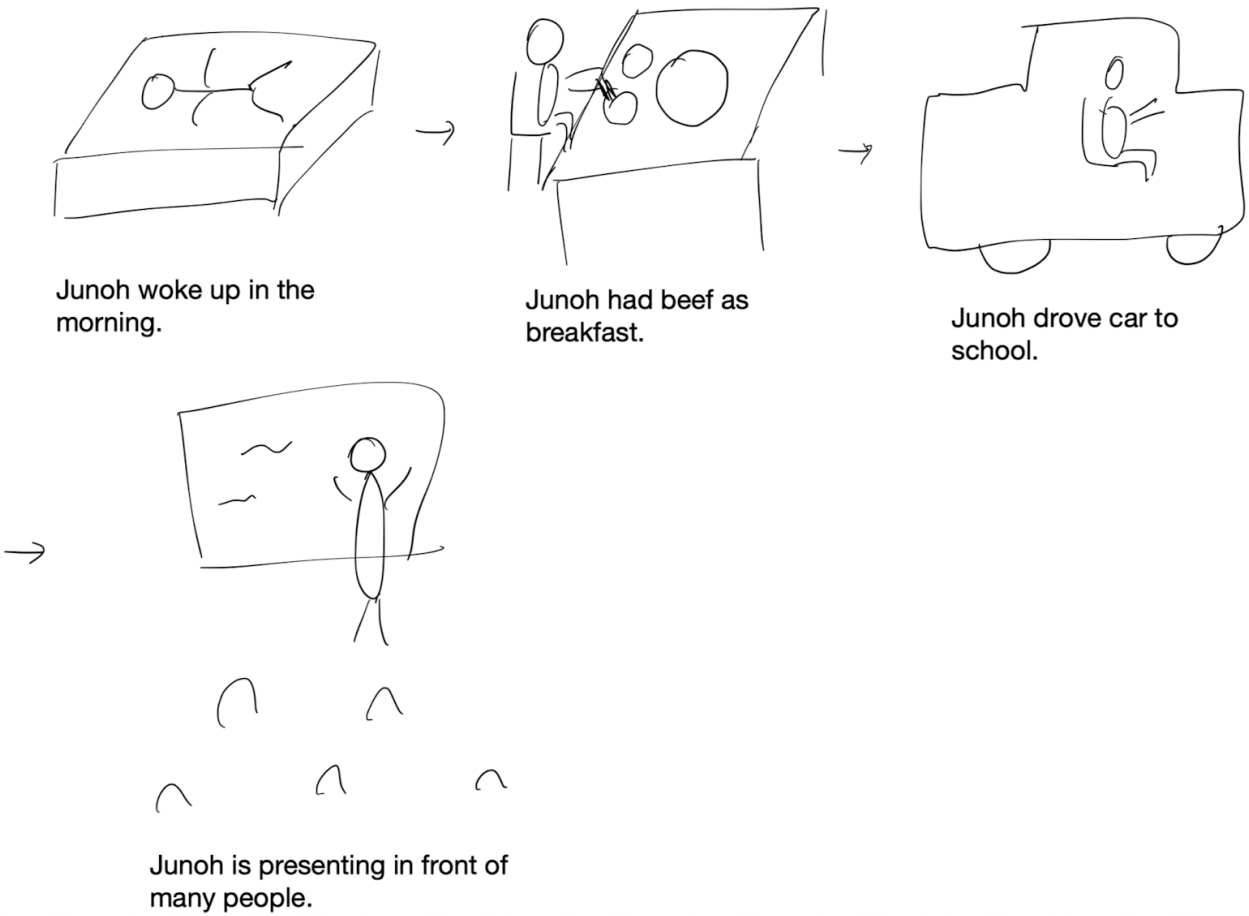
We can use both of them. We can give finer snapshots for complicated motions, and coaser snapshots with texts for simple motions.



How can we describe video?

Texts and Images

I tried one for myself, but . . .



How can we describe video?

Texts and Images

To conclude,

	Apearance	Motion	Background	Costs
Texts	△	△	△	Low
Images	◯	△	◯	High
Texts and Images	◯	△	◯	Mid $\sim$ High

Generating Videos from an Image and Texts

Generating is the opposite of describing, and it is even harder.

Many video diffusion models generate a video conditioned on an **image** and **texts**, and they have shown pretty good qualities. However, are they expected videos? Do objects move just as you imagined?

We need more conditions to generate desired videos!

Image + Text Image + Text + something

DragNUWA

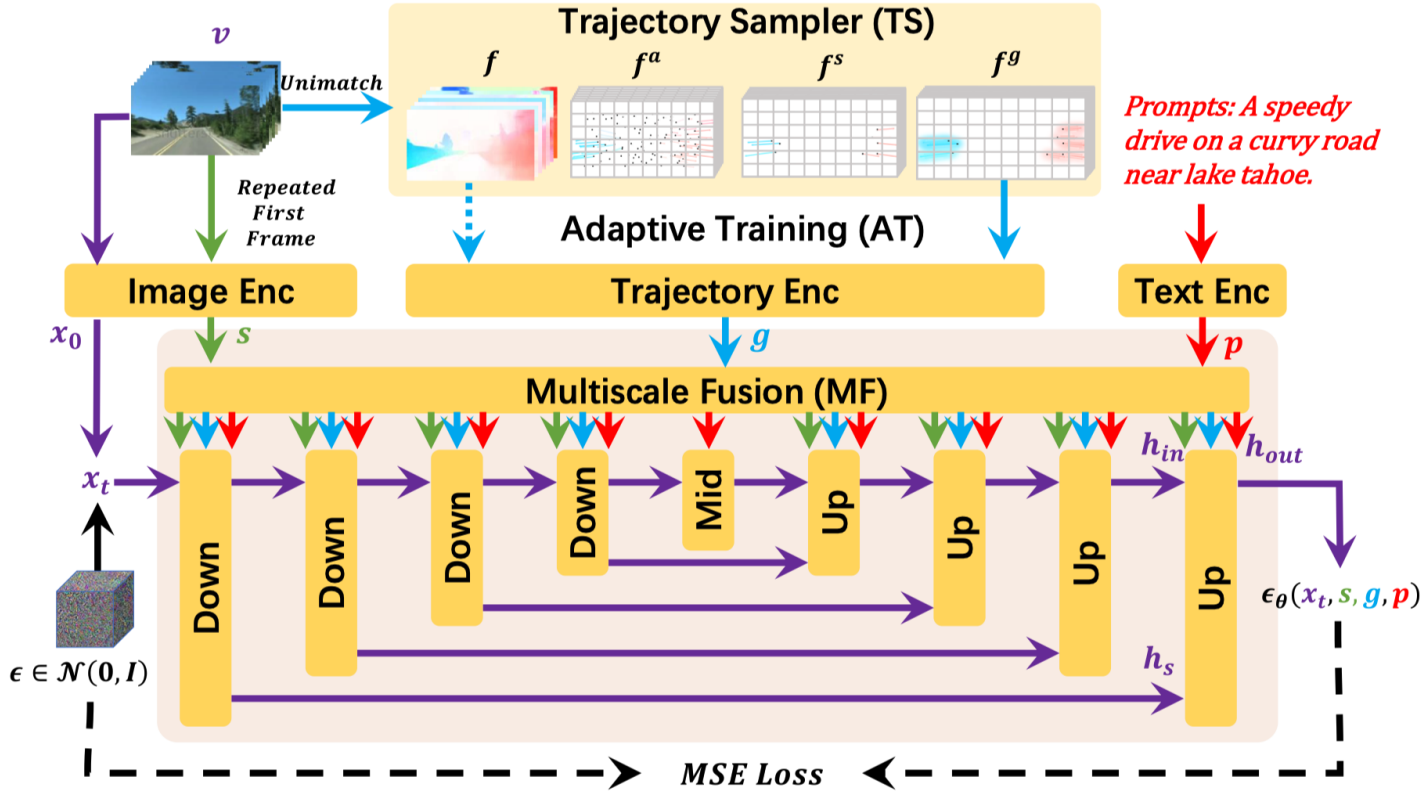
Overview

An image, texts, and **trajectories** to control videos.

DragNUWA

Overview

The method is simple; each condition is encoded and then integrated.



DragNUWA

Preparing Dataset

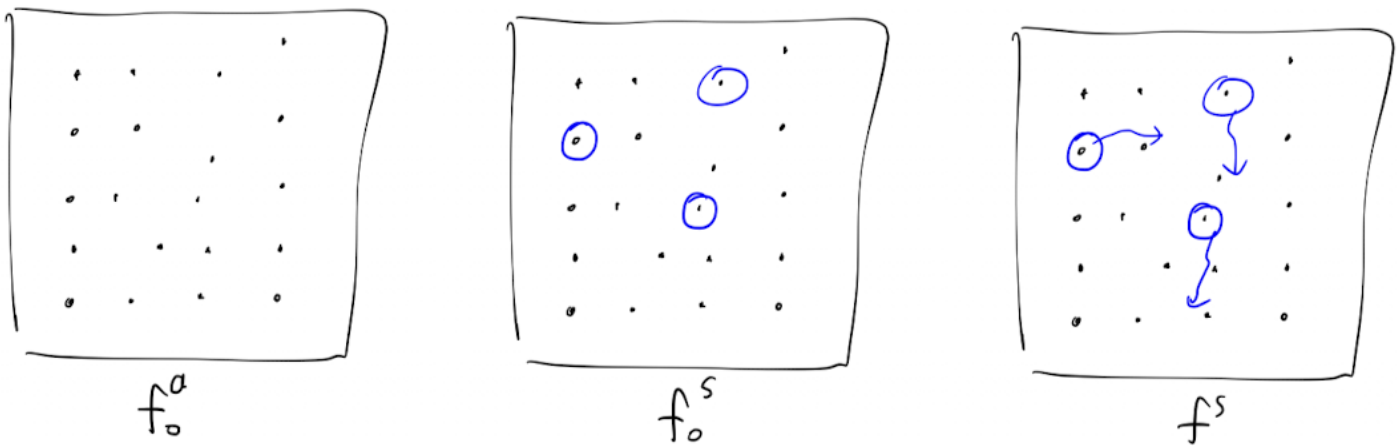
The training data usually consists of (video, text) pairs. Therefore, they **extract trajectories** from videos.

Trajectory Sampler (TS) samples trajectories from video optical flow.

1. Obtain optical flow map $f \in \mathbb{R}^{(L-1)\times C\times H\times W}$ from a video.
2. Obtain slightly sparse map f_o^a by distributing anchor points with interval of λ with jittering.
3. Obtain sparse map f_o^s by sampling n tracking points from f_o^a, where $n \sim U[1, N]$.
   - Can handle various number of trajectories.
4. Obtain full trajectory f^s by iteratively track the trajectories by updating tracking points.
5. Obtain smooth trajectory map f^g by applying Gaussian filter to f^s.

DragNUWA

Preparing Dataset



DragNUWA

Training

Video, image, texts and trajectories are encoded to $\mathbb{R}^{L \times c_{\text{sth}} \times h \times w}$.

- Video (x_t) and Image (s) : Encoder from [Rombach et al., 2022]
- Texts (p) : CLIP encoder [Radford et al., 2021]
- Trajectory (g) : a series of convolution layers

Then, image embedding s, prompt embedding p and trajectory embedding g is linearly fused. (Multiscale fusion)

DragNUWA

Training

- Randomly omits texts, image, and trajectories before multiscale fusion.
- Adaptive training
  1. $\epsilon_\theta(x_t, p, s, f)$: train with dense map f.
  2. $\epsilon_\theta(x_t, p, s, g)$: train with sparse and smooth map g.

Boximator

Overview

An image, texts, and **bounding boxes (with or without trajectories)** to control videos.

The kitten is hiding herself into the cup.

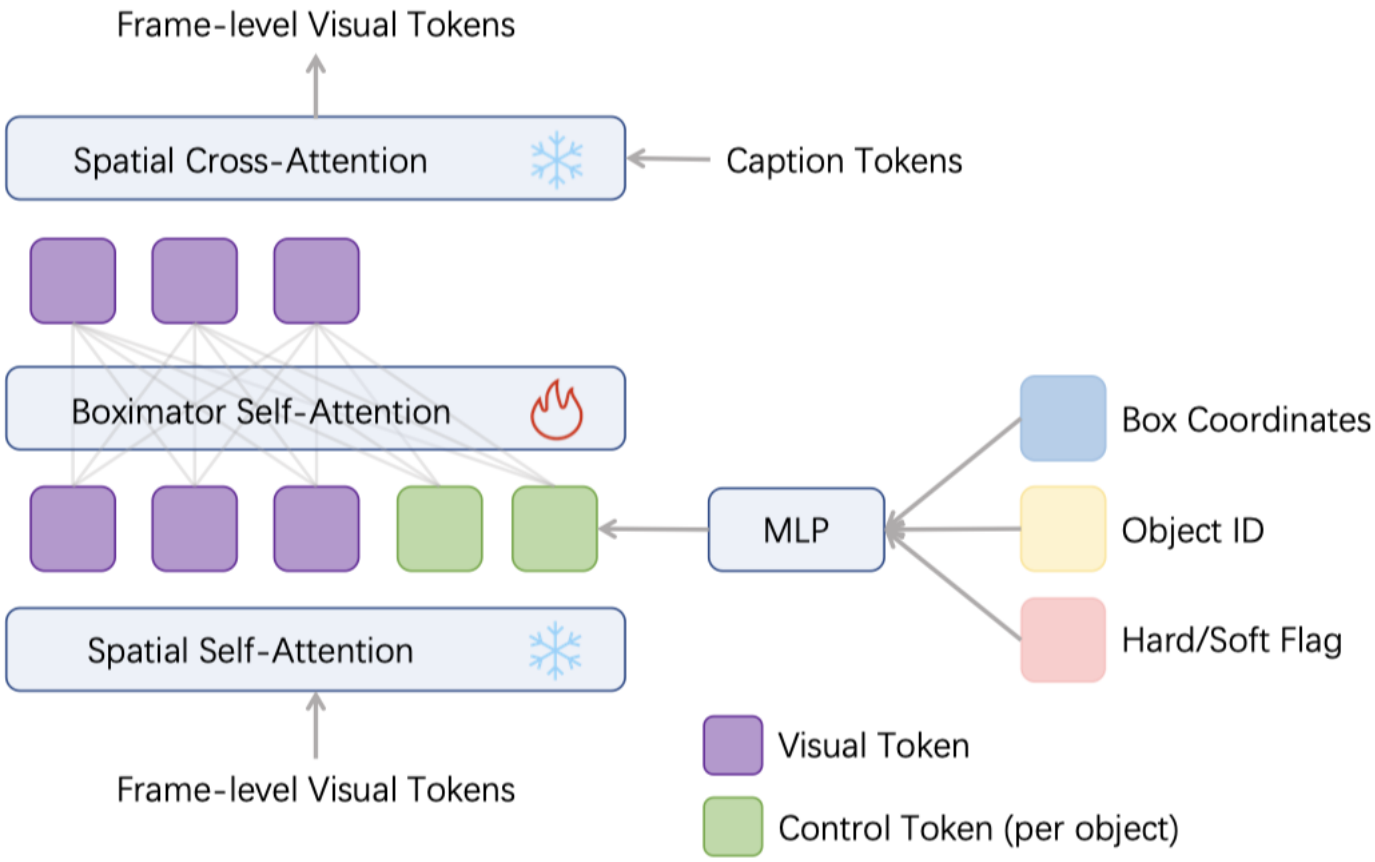
A bird with grey, red and yellow feathers flys away.

Drone flying over New Zealand beach.

Boximator

Overview

Control tockens include information of box coordinates, object ID, and hard/soft box indicator.



Boximator

Preparing Dataset

From videos, they extract image description and

1. Sample dynamic subset of WebVid.
2. Generate image description of the first frame.
3. Use nouns for object tracking to generate bounding boxes.
4. If bounding box does not fall in cropped area, it is projected.
   - This enables disappearing or appearing motions.

Boximator

Training

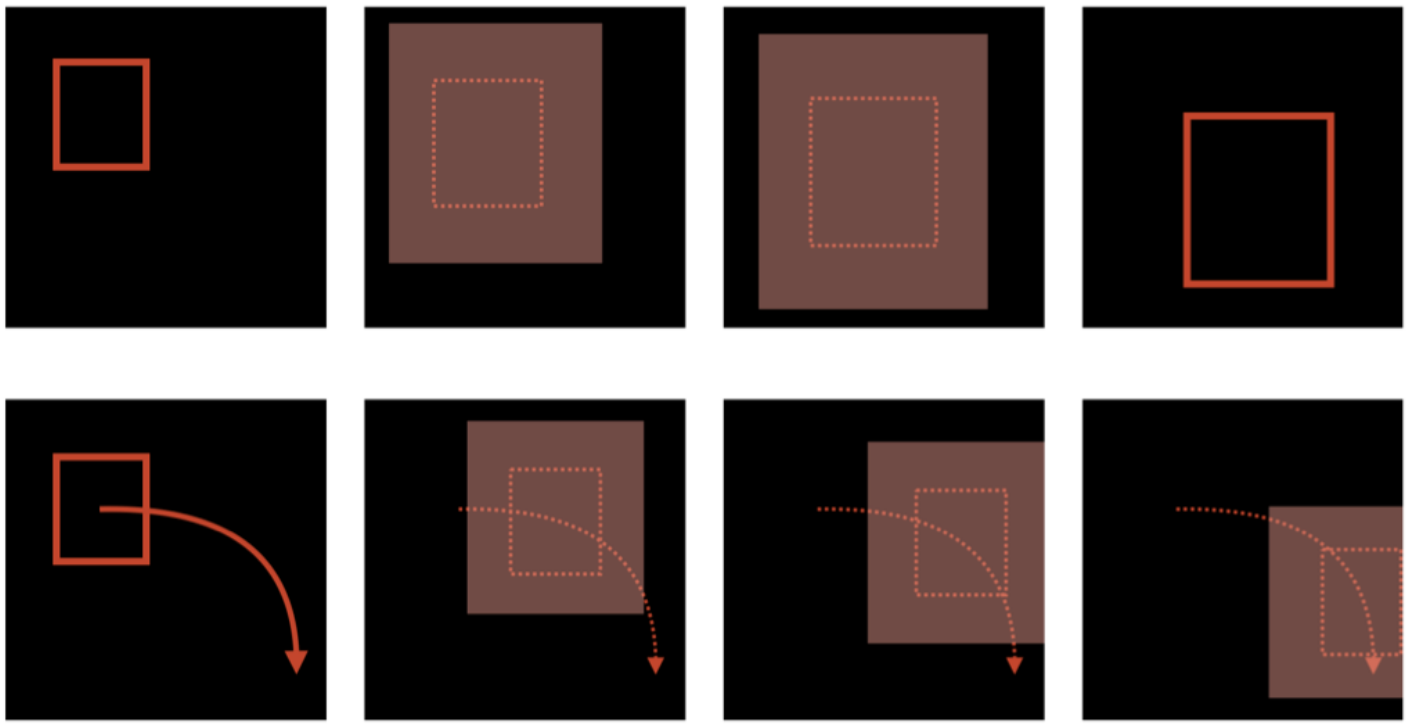
- Self-tracking
  - Diffusion model generates colored bounding boxes.
- Without self-tracking
  - Train model to stop generating bounding boxes.

1. Self-tracking with hard bounding boxes.
2. Self-tracking with hard (20%) and soft (80%) bounding boxes.
3. Without self-tracking with hard (20%) and soft (80%) bounding boxes.

Boximator

Inference

1. User select first and last bounding boxes, or trajectories.
2. Between first and last frame, Boximator generates soft boxes for motion control.



Reference I

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021).
Learning transferable visual models from natural language supervision.
In *PMLR*.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022).
High-resolution image synthesis with latent diffusion models.
In *CVPR*.

Wang, J., Zhang, Y., Zou, J., Zeng, Y., Wei, G., Yuan, L., and Li, H. (2024).
Boximator: Generating rich and controllable motions for video synthesis.
arXiv preprint arXiv:2402.01566.

Yin, S., Wu, C., Liang, J., Shi, J., Li, H., Ming, G., and Duan, N. (2023).
Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory.