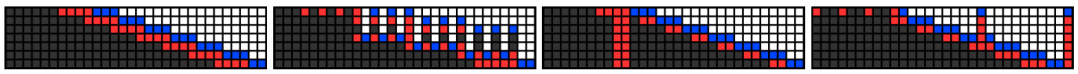
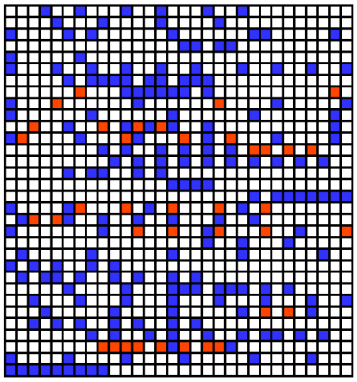
Flexible Diffusion Modeling of Long Videos

- アブスト
  - 様々な現実的な環境における長時間のvideo completion(動画補間や予測)を可能にしたDDPMを提案
  - 推論時には、他のビデオフレームを条件付けて、任意のビデオフレームを生成できる
  - 時系列的に真っ当な25分もの動画を生成することが可能になった。
- 概要
  - K個のフレームで学習させたモデルから効率よくN>K個のフレームを生成したら良いのかというのが動画生成における問題である
    - 自己回帰 - (a)
    - 一つのモデルで4フレームに一回フレームを生成して、もう一つのモデルでその間を埋める - (b)
      - 二つのモデルを学習させる
  - DDPMに基づくflexibleなモデル構造を提案
    - DDPM + temporal attention
    - 任意の過去もしくは未来のフレームを条件付けすることができる - (c), (d)



(a) Autoregressive. (b) Two temporal res. (c) Long-range (ours). (d) Hierarchy-2 (ours).

      - 任意に条件付けフレームと予測フレームを選択



Algorithm 2 Sampling training tasks $\mathcal{X}, \mathcal{Y} \sim u(\cdot)$ given N, K.

1: $\mathcal{X} := \{\}; \mathcal{Y} := \{\}$
2: **while** True **do**
3: $\quad n_{\text{group}} \sim \text{UniformDiscrete}(1, K)$
4: $\quad s_{\text{group}} \sim \text{LogUniform}(1, (N-1)/n_{\text{group}})$
5: $\quad x_{\text{group}} \sim \text{Uniform}(0, N - (n_{\text{group}} - 1) \cdot s_{\text{group}})$
6: $\quad o_{\text{group}} \sim \text{Bernoulli}(0.5)$
7: $\quad \mathcal{G} := \{\lfloor x_{\text{group}} + s_{\text{group}} \cdot i \rfloor | i \in \{0, \ldots, n_{\text{group}} - 1\}\} \setminus \mathcal{X} \setminus \mathcal{Y}$
8: $\quad$ **if** $|\mathcal{X}| + |\mathcal{Y}| + |\mathcal{G}| > K$ **then**
9: $\quad\quad$ **return** `set2vector`$(\mathcal{X})$, `set2vector`$(\mathcal{Y})$
10: $\quad$ **else if** $|\mathcal{X}| = 0$ **or** $o_{\text{group}} = 0$ **then**
11: $\quad\quad \mathcal{X} := \mathcal{X} \cup \mathcal{G}$
12: $\quad$ **else**
13: $\quad\quad \mathcal{Y} := \mathcal{Y} \cup \mathcal{G}$

Figure 4: **Left:** Samples from $u(\mathcal{X}, \mathcal{Y})$ with video length $N = 30$ and limit $K = 10$ on the number of sampled indices. Each row shows one sample and columns map to frames, with frame 1 on the left and frame N on the right. Blue and red denote latent and observed frames respectively. All other frames are ignored and shown as white. **Right:** Pseudocode for drawing these samples. The while loop iterates over a series of regularly-spaced groups of latent variables. Each group is parameterized by: the number of indices in it, n_{group}; the spacing between indices in it, s_{group}; the position of the first frame in it, x_{group}, and an indicator variable for whether this group is observed, o_{group} (which is ignored on line 10 if $\mathcal{X}$ is empty to ensure that the returned value of $\mathcal{X}$ is never empty). These quantities are sampled in a continuous space and then discretized to make a set of integer coordinates on line 7. The process repeats until a group is sampled which, if added to $\mathcal{X}$ or $\mathcal{Y}$, will cause the number of frames to exceed K. That group is then discarded and $\mathcal{X}$ and $\mathcal{Y}$ are returned as vectors. The FDM's training objective forces it to work well for any $(\mathcal{X}, \mathcal{Y})$ pair from this broad distribution.

- モデル構造
  - ガウスノイズを加えていき、条件付けフレームとガウスノイズからだんだんぼかしを消すように学習

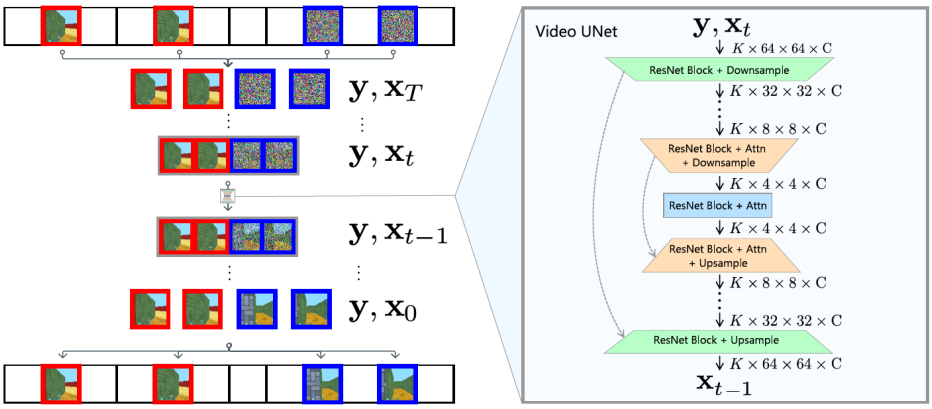
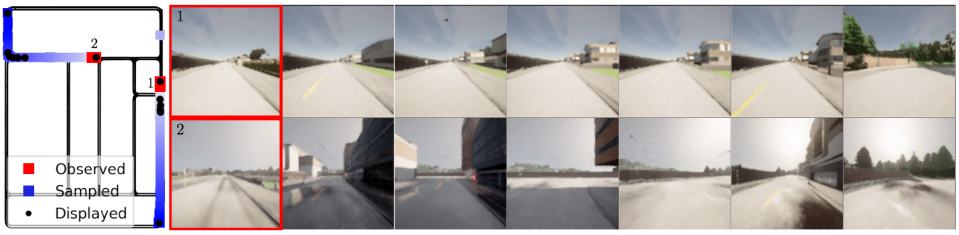


Figure 3: **Left:** Our DDPM iteratively transforms Gaussian noise $\mathbf{x}_T$ to video frames $\mathbf{x}_0$ (shown with blue borders), conditioning on observed frames $\mathbf{y}$ (red borders) at every step. **Right:** The U-net architecture used within each DDPM step. It computes $\epsilon_\theta(\mathbf{x}_t, \mathbf{y}, t)$, with which the Gaussian transition $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is parameterized.

- データセット
  - CARLA Town01 Dataset
    - 408 training, 100 test

        - 1000 frames with 128×128 resolution
    - 生成したフレームから車の位置(x,y)を推定するようなNNを学習
        - これによってどのような経路を生成したか確認することができる



        - また生成フレームから車の速度も定期的に計算することができる
            - 生成がうまくいかないと、フレームが飛んだりし、速度が異常に大きい値をとる
                - どのくらい閾値を超えたか
                    - outlier percentage (OP)
                - 元の動画の速度の分布との距離
                    - Wasserstein Distance (WD)
- 実験
    - データセット
        - GQN-Mazes
            - 300フレーム
        - MineRL
            - 500フレーム
        - CARLA Town01 dataset
            - 1000フレーム
        - 推論時にはいずれも36フレーム条件づけて生成
        - 評価指標は主にFVD
            - GQN-Mazesのaccuracyはどのくらい正解の部屋に訪れたかの指標

- 結果

Model	Sampling scheme	GQN-Mazes FVD	GQN-Mazes Accuracy	MineRL FVD	CARLA Town01 FVD	CARLA Town01 WD	CARLA Town01 OP
CWVAE [26]	CWVAE	837 ± 8	82.6 ± 0.5	1573 ± 5	1161	0.666	44.4
TATS [11]	TATS	163 ± 2.6	77.0 ± 0.8	807 ± 14	329	1.648	42.4
VDM [16]	VDM	66.7 ± 1.5	77.8 ± 0.5	271 ± 8.8	169	0.501	16.9
FDM (ours)	Autoreg	86.4 ± 5.2	69.6 ± 1.3	281 ± 10	222	0.579	0.51
	Long-range	64.5 ± 1.9	77.0 ± 1.4	$\mathbf{267 \pm 4.0}$	213	0.653	**0.47**
	Hierarchy-2	$\mathbf{53.1 \pm 1.1}$	82.8 ± 0.7	275 ± 7.7	120	0.318	3.28
	Hierarchy-3	53.7 ± 1.9	$\mathbf{83.8 \pm 1.1}$	311 ± 6.8	149	0.363	4.53
	Ad. hierarchy-2	55.0 ± 1.4	83.2 ± 1.3	316 ± 8.9	**117**	**0.311**	3.44

  - VDM
    - frameskip-4 model + frameskip-1 model
  - FDM

Sampling schemes Before describing the sampling schemes we explore experimentally, we emphasize that the relative performance of each is dataset-dependent and there is no single best choice. A central benefit of FDM is that it can be used at test-time with different sampling schemes without retraining. Our simplest sampling scheme, **Autoreg**, samples ten consecutives frames at each stage conditioned on the previous ten frames. **Long-range** is similar to Autoreg but conditions on only the five most recent frames as well as five of the original 36 observed frames. **Hierarchy-2** uses a multi-level sampling procedure. In the first level, ten evenly spaced frames spanning the non-observed portion of the video are sampled (conditioned on ten observed frames). In the second level, groups of consecutive frames are sampled conditioned on the closest past and future frames until all frames have been sampled. **Hierarchy-3** adds an intermediate stage where several groups of variables with an intermediate spacing between them are sampled. We include adaptive hierarchy-2, abbreviated **Ad. hierarchy-2**, as a demonstration of a sampling scheme only possible with a model like FDM. It samples the same frames at each stage as Hierarchy-2 but selects which frames to condition on adaptively at test-time with a heuristic aimed at collecting the maximally diverse set of frames, as measured by the pairwise LPIPS distance [41] between them.

- 欠点
  - 300フレームの生成に16分かかる