ExtDM: Distribution Extrapolation Diffusion Model for Video Prediction

https://zzcheng.top/ExtDM

Zhicheng Zhang[1,2,†] Junyao Hu[1,2,†] Wentao Cheng[1,‡] Danda Paudel[3,4] Jufeng Yang[1,2]

[1]VCIP & TMCC & DISSec, College of Computer Science, Nankai University

[2]Nankai International Advanced Research Institute (SHENZHEN· FUTIAN)

[3]Computer Vision Lab, ETH Zurich [4]INSAIT, Sofia University

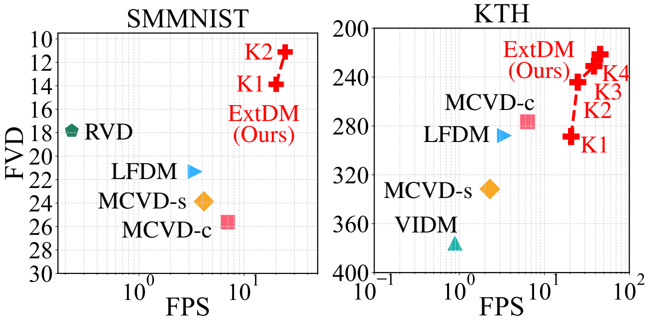
gloryzzc6@sina.com,hujunyao@mail.nankai.edu.cn,{wentaocheng,yangjufeng}@nankai.edu.cn,danda.paudel@insait.ai

Abstract

Video prediction is a challenging task due to its nature of uncertainty, especially for forecasting a long period. To model the temporal dynamics, advanced methods benefit from the recent success of diffusion models, and repeatedly refine the predicted future frames with 3D spatiotemporal U-Net. However, there exists a gap between the present and future and the repeated usage of U-Net brings a heavy computation burden. To address this, we propose a diffusion-based video prediction method that predicts future frames by extrapolating the present distribution of features, namely ExtDM. Specifically, our method consists of three components: (i) a motion autoencoder conducts a bijection transformation between video frames and motion cues; (ii) a layered distribution adaptor module extrapolates the present features in the guidance of Gaussian distribution; (iii) a 3D U-Net architecture specialized for jointly fusing guidance and features among the temporal dimension by spatiotemporal-window attention. Extensive experiments on five popular benchmarks covering short- and long-term video prediction verify the effectiveness of ExtDM.

1. Introduction

Video prediction is a long-standing and challenging task in computer vision. It aims to forecast future frames of a video like humans and serves as a key component of intelligent decision-making systems [48]. Predicting the possible future with pixel-level details can benefit trustable decisions, which is especially crucial for scenarios involving human security. Take an instance, considering an automatic vehicle driving at a crowded intersection, an important problem is to predict how people will move. Therefore, various downstream applications have been developed, ranging from autonomous driving [1,7,83], robotic navigation [16,33,84], artistic design [19,42,56,57,71], and video understanding [80,81,85,86].



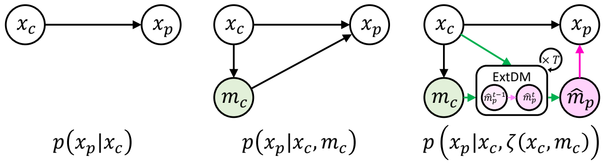
(a) Short-term Video Prediction (b) Long-term Video Prediction

Figure 1. **A comparison of quality and speed of SOTA diffusion models** for short-term and long-term video prediction on SMMNIST and KTH, respectively. We report FVD as well as FPS. Note that the FPS axis is in the log scale.

Advanced works in video prediction [20,22,64,66] propose to capture the dynamic change in the video [2,18,36,60]. Direct methods [5,27] (Fig. 2 (a)) only take RGB frames as input and find that the video prediction problem is difficult to be solved due to its inherent high complexity (*i.e.*, estimating the posterior probability of $p(\boldsymbol{x}_p|\boldsymbol{x}_c)$). Thus, in-context learning methods [55,61] (Fig. 2 (b)) incorporates semantic cues as key information, where motion cues requiring no extra model can be easily introduced as implicit guidance $p(\boldsymbol{x}_p|\boldsymbol{x}_c, \boldsymbol{m}_c)$. A common characteristic of these methods is that they have excellent predictive capacity in a short time span but lack accuracy over a longer period, leading to counterfactual results like videos that fade to grey. The reason behind this is that these methods take no deterministic cues for the future, which has a huge gap.

Capturing future cues is challenging due to the uncertainty of the future. It is required to understand high-level spatiotemporal correlation and model potential proposals for future [21]. Several recent attempts [14,24,62] embrace the popular video diffusion models [26,50] and try to reformulate estimating the future distribution as a serial denoising process. Attributed to its ODE formula, the future frames can be obtained through a series of chained Markovian steps. However, this incurs a significant computational burden due to the repeated usage of spatiotemporal U-Net.

† Equal Contribution. ‡ Corresponding Author.



(a) *Direct* (b) *In-context Learning* (c) *Extrapolation* (Ours)

Figure 2. **Comparison between ExtDM and other video prediction methods.** The graphical model shows the prediction process with motion cues. $\boldsymbol{x}_c, \boldsymbol{x}_p$ indicate the condition frames and the predicted ones, respectively. $\boldsymbol{m}_c, \hat{\boldsymbol{m}}_p$ represent the motion cues from the condition frames and the ones predicted by our distribution extrapolation diffusion model ζ, respectively. $\hat{\boldsymbol{m}}_p^{\{t-1,t\}}$ are the intermediate results during extrapolating motion cues.

As a result, the processing speed becomes sluggish, in terms of single-digit frames per second (FPS), as shown in Fig. 1.

In this work, we provide a novel view that casts the video prediction problem as extrapolating the deterministic motion cues from present to future (*i.e.*, $\hat{\boldsymbol{m}}_p = \zeta(\boldsymbol{x}_c, \boldsymbol{m}_c)$, as shown in Fig. 2 (c)). Our insight is that incorporating future cues to generate corresponding frames is much easier than hallucinating them from scratch. By gradually estimating the shifted distribution from the present one, ExtDM can maintain temporal consistency and avoid drastic performance drops for long-term video prediction. As a byproduct of decoupling motion extrapolation and frame prediction, ExtDM can not only produce desired outcomes, but also enable us to customize future motion and generate potential proposals for stochastic events. Besides, the compacted resolution of motion cues cut down the computation cost raised by the prediction model.

To extrapolate future motion cues, we first extract motion cues from condition frames with a lightweight motion autoencoder. Then, we propose a probability diffusion model that extrapolates the motion cues via a series of Markovin steps. We make use of the present features to estimate the parameter of the video distribution. To account for temporal dynamics, we develop a layered distribution adapter that predicts the corresponding parameters over time. This allows us to easily generate future features based on the estimated distribution. With the extrapolated features, we employ a sparse spatiotemporal window U-Net to fuse it with plain features to refine the predicted future cues. The predicted future cues are ultimately used for reconstructing the video frames as the extrapolated ones.

Our contributions are three-fold: (1) We propose a distribution extrapolation diffusion model that forecasts the future frames by extrapolating from the present frames. (2) We propose an efficient video prediction method that includes compression and reconstruction. By imitating future motion cues, our approach can create tailored proposals for stochastic events. (3) Extensive experiments on five popular benchmarks verify the effectiveness of our method for both short- and long-term video prediction.

2. Related Work

Video Prediction forecasts the future frames at the pixel level [4, 9, 10, 23, 70, 75, 79] and models the change among frames [14, 34, 44, 52, 58, 65, 76]. It is crucial for downstream applications such as representation [11, 31, 63, 69], detection [37–39], segmentation [12,29,30,88], and restoration [43,91–93]. In earlier work, [15,36] propose stochastic variational inference-based methods that explicitly extract spatial and temporal information. PRNN [66] constructs spatiotemporal LSTM, SLAMP [3] learns the prior distribution from the appearance and optical flow, and MOSO [55] decouples the frames to motion, scene and object tensors.
Video Diffusion Models learn to transform the Gaussian noise distribution to a video-related distribution [8, 41, 47, 74, 78]. This process depends on multi-round condition-guided denoise iteration. VDM [26] firstly proves the feasibility for DM to complete video tasks. RVD [73] proposes a DM that predicts the residual error of the next video frame every time. MCVD [62] implements a general multiple-inputs-multiple-output VDM based on 2D convolution by compressing dimension. RaMViD [28] introduces random masks and constructs a 3D convolution VDM. LVDM [25] uses a 3D autoencoder and hierarchical mechanism to generate any length of the video in latent space.

Step into deep era [35, 68, 87, 89, 90], various methods have made efforts to predict the future as mentioned above, modeling long-range temporal dynamics still remains challenging due to the inherent shortcomings of each solution. Direct methods bring high computation costs and, thus are hard to deploy in low-resource devices. In-context learning methods rely on the semantic cues inferred from current frames, which have a gap towards the ones in the future.

3. Methodology

The pipeline of the proposed ExtDM is illustrated in Fig. 3. Given a series of condition frames $\boldsymbol{x}_c$, the objective of ExtDM is to forecast the future frames $\boldsymbol{x}_p$ in the video by fully exploiting the appearance and motion cues. Let the lengths of $\boldsymbol{x}_c, \boldsymbol{x}_p$ be u, v, respectively. The workflow of our method can be summarized into three parts: (i) The motion autoencoder compression (Sec. 3.1), (ii) Distribution extrapolation diffusion model (Sec. 3.2) , and (iii) The motion autoencoder reconstruction (Sec. 3.1). The encoder of motion autoencoder projects the condition frames $\boldsymbol{x}_c$ into a series of motion cues $\boldsymbol{m}_c$ (*i.e.*, optical flows and occlusion maps). Then, layered distribution adaptor extrapolates the features into the future via a stack of the Gaussian processions. SpatioTemporal-Window (STW) U-Net takes future features as reference through attention, resulting in generation of future motion cues $\hat{\boldsymbol{m}}_p$. Finally, the decoder of motion autoencoder reconstructs the future frames $\boldsymbol{x}_p$ from the predicted motion cues $\hat{\boldsymbol{m}}_p$ and the condition frames $\boldsymbol{x}_c$.

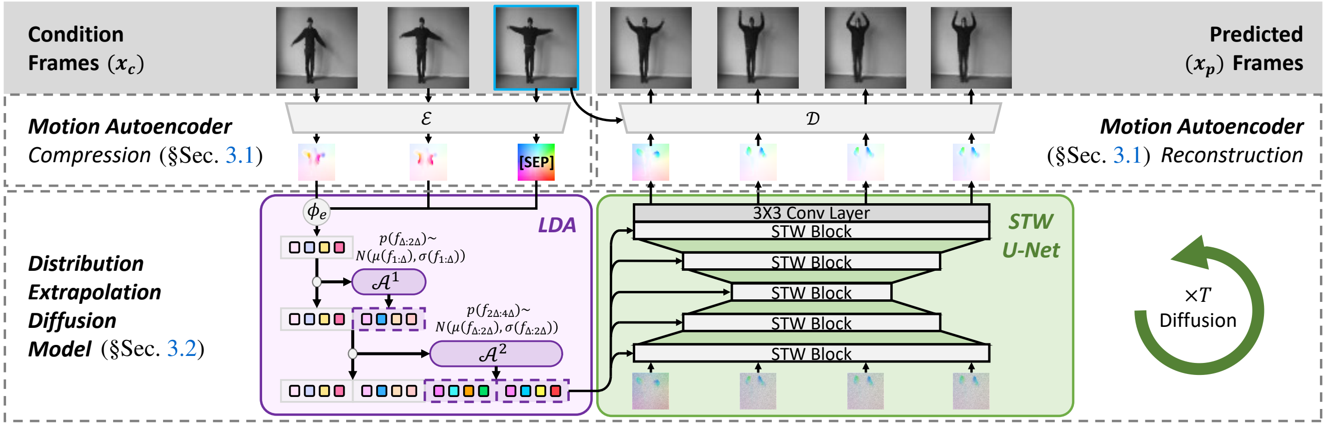

Condition Frames (x_c)
Predicted (x_p) Frames
Motion Autoencoder Compression (§Sec. 3.1)
$\mathcal{E}$
[SEP]
$\mathcal{D}$
Motion Autoencoder (§Sec. 3.1) Reconstruction
Distribution Extrapolation Diffusion Model (§Sec. 3.2)
ϕ_e
LDA
$p(f_{\Delta:2\Delta})\sim N(\mu(f_{1:\Delta}),\sigma(f_{1:\Delta}))$
$\mathcal{A}^1$
$p(f_{2\Delta:4\Delta})\sim N(\mu(f_{\Delta:2\Delta}),\sigma(f_{\Delta:2\Delta}))$
$\mathcal{A}^2$
3X3 Conv Layer
STW Block
STW Block
STW Block
STW Block
STW Block
STW U-Net
×T Diffusion

Figure 3. **Pipeline of ExtDM.** ExtDM consists of three main components: Motion autoencoder constructs a bijection transform between the pixel space and motion space via compression and reconstruction. The layered distribution adaptor extrapolates the features of future frames as a shifted distribution derived from condition frames. Furthermore, the built STW U-Net takes the extrapolated feature as guidance and conducts sparse and stride attention among spatiotemporal dimensions for encouraging feature interactions.

Now, to lay the groundwork for predicting future outcomes, we begin by establishing a bijection transformation that includes two mapping functions: $\boldsymbol{x}_c \rightarrow \boldsymbol{m}_c$ and $\hat{\boldsymbol{m}}_p \rightarrow \boldsymbol{x}_p$.

3.1. Motion Autoencoder

As discussed above, motion provides deterministic cues for reasoning the future. To this end, we compress the video by a lightweight motion autoencoder which conducts a bijection transformation between motion cues and video frames. Built on top of lightweight autoencoder architecture [32], the motion autoencoder consists of two stages: the encoder $\mathcal{E}$ extracts motion cues from frames, and the decoder $\mathcal{D}$ reconstructs video frames from the motion cues.

• **Motion Autoencoder Compression** In order to extract motion cues from a series of condition frames, encoder $\mathcal{E}(\cdot,\cdot)$ estimates the optical flow and occlusion map between video frames in a pair-wise manner. For condition frames $\boldsymbol{x}_c = \left\{\boldsymbol{x}_i \in \mathbb{R}^{CHW} \mid i = 1, \dots, u\right\}$ with a length of u, we extract the motion cues between each condition frame $\boldsymbol{x}_i$ and the last condition frame $\boldsymbol{x}_u$ (*i.e.*, the keyframe that will be used for reconstruction). These paired frames are fed into the encoder to estimate the motion correlation between them, including the optical flow w_i and its corresponding occlusion map o_i as

$$\boldsymbol{m}_c = \left\{\boldsymbol{m}_i \in \mathbb{R}^{3hw} \mid \boldsymbol{m}_i = \mathcal{E}(\boldsymbol{x}_i, \boldsymbol{x}_u) = \begin{bmatrix} w_i \\ o_i \end{bmatrix}\right\}. \quad (1)$$

Notably, the motion cues of the keyframes are replaced with a learnable token. To represent the pixel offsets, we compute the flow w_i from the condition frame to the keyframe, resulting in a size of $2 \times h \times w$ to describe vertical and horizontal movements. To model challenging cases such as occluded backgrounds, we estimate the occlusion map o_i which indicates the degree of occlusion on a scale from 0 to 1 and has a size of $1 \times h \times w$. Here, $h = H/S, w = W/S$, and S is the downsampling factor.

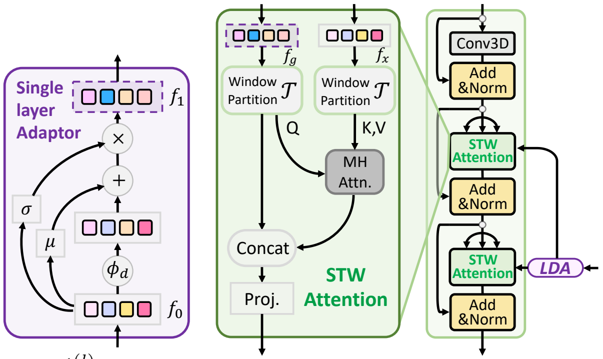
• **Motion Autoencoder Reconstruction** With the extrapolated motion cues for the future $\hat{\boldsymbol{m}}_p = \{\hat{\boldsymbol{m}}_j \in \mathbb{R}^{3hw} | j = 1, \dots, v\}$ and the keyframe, the decoder $\mathcal{D}(\cdot,\cdot)$ reconstructs the future frames in a pair-wise manner, similar to the encoder. We pair the keyframe and the predicted motion cue of the j^{th} future frame. The latent representation of condition frames $\boldsymbol{z}_u$ is first warped with the guidance of flow w_j. Considering the occlusion, the warped representation is further filtered by incorporating the occlusion map of each predicted future frame o_j as $o_j \odot \mathcal{W}(\boldsymbol{z}_u, w_j)$. The representation is further fed into network $\mathcal{G}$ for inpainting the occluded area. Here, $\mathcal{W}(\boldsymbol{z}, w)$ is the warp operation for feature $\boldsymbol{z}$ guided by flow w, and $\odot$ is the element-wise product. The reconstructed frames are finally obtained as

$$\boldsymbol{x}_p = \{\boldsymbol{x}_j \in \mathbb{R}^{3HW} | \boldsymbol{x}_j = \mathcal{D}(\hat{\boldsymbol{m}}_j, \boldsymbol{x}_u) = \mathcal{G}(o_j \odot \mathcal{W}(\boldsymbol{z}_u, w_j))\}. \quad (2)$$

Considering the problem at hand, we proceed to the subsequent stage of extrapolating the future motion cues through the estimation of distribution shift $\boldsymbol{m}_c, \boldsymbol{x}_c \rightarrow \hat{\boldsymbol{m}}_p$.

3.2. Distribution Extrapolation Diffusion Model

Advanced methods exploit the correlation alongside the temporal dimension implicitly by temporal attention or 1D temporal convolutional network to generate future frames. While these approaches are effective in generating the feature of future frames, they encode the previous frame into the network to predict the future one, which overlooks the distribution prior among frames and generates counterfactual samples due to uncertainty of the future. In contrast, we propose a distribution extrapolation diffusion model to extrapolate the motion cues $\hat{\boldsymbol{m}}_p$ through a series of backward (denoising) steps. Based on the assumption of Gaussian mixture model, we design a layered distribution adaptor



(a) $\mathcal{A}^{(l)}$ in LDA (b) STW Attention

Figure 4. **Illustration of the detailed structure** of (a) single layer adaptor and (b) spatiotemporal window block. For more details please refer to Alg. 1.

to model the shifted distribution of future features causally and further introduce spatiotemporal attention to fuse the extrapolated feature and the plain feature.

With the extracted motion cues $\boldsymbol{m}_c$ and appearance feature from latent representation $\boldsymbol{z}_c$, our video diffusion model consists of a forward function $\{q_t\}_{t\in[0,1]}$ to add a series of noise into the future frames $\boldsymbol{m}_p^1 \sim q_1(\boldsymbol{m}_p^1)$, and a backward function $\{p_t\}_{t\in[0,1]}$ to predict the future frames from the Gaussian noise $p_1(\boldsymbol{m}_p^0) := \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ by the proposed *Spatiotemporal Window U-Net* $\boldsymbol{\epsilon}_\theta(\boldsymbol{m}_p^t, c)$. We take both the appearance feature (latent representation $\boldsymbol{z}_c$) and the motion feature (motion cues $\boldsymbol{m}_c$) as guidance c. To bridge the difference between present and future, we extrapolate the guidance from condition frames f_c into the future one f_p by the proposed *Layered Distribution Adaptor*.

• Layered Distribution Adaptor Due to the challenge of forecasting the future, there is a huge gap between future and present. To overcome this challenge, we represent the video frames as the same distribution. This enables us to anticipate future samples using an autoregressive adapter. We propose a layered distribution adaptor (LDA) that initially 'encode' the condition frames to estimate the distribution parameter and 'inference' the future frame using distribution sampling for prediction. Unlike existing temporal correlation methods that estimate the future frame implicitly, LDA introduces the distribution prior as a constraint. To better fit the needs of extrapolation, the types of input features of LDA include latent representation $\boldsymbol{z}_c$, motion cues $\boldsymbol{m}_c$, and so on. The pseudocode of LDA is in Alg. 1, and the details are given in the next paragraph.

Given the features from condition frames with a length of Δ, LDA aims to produce an extrapolated feature of future frames. The input features f_c are first fed into projector ϕ_e to exploit temporal correlation among condition frames, and then are extrapolated into the future in a multi-layered manner. For the l^{th} layer, we predict future frames $\widehat{f}_{\Delta:2^l\Delta}$ from the present ones $f_{1:\Delta}$ via a single-layer adaptor $\mathcal{A}^{(l)}$:

Algorithm 1 LDA: Pytorch-style Pseudocode

```
# f: input feature (N C T H W)
# phi_e: encoding layer
# phi_d: decoding layers
# L: number of layers

f = phi_e(f) # encoding condition frames
for l in range(L):
    r = f
    mu, var = est(f) # Gaussian prior
    f_h = (f - mu) / std
    mu = m_est(f_h) + mu
    var = (1 + v_est(f_h)) * var
    f_h = phi_d[l](f_h) # inferencing future frames
    f = f_h * var + mu
    f = torch.cat([r,f], dim=2)

# distribution estimation
def est(f, eps=1e-5):
    f_var = f.view(N, C, T, -1).var(dim=3) + eps
    f_std = f_var.sqrt().view(N, C, T)
    f_mean = f.view(N, C, T, -1).mean(dim=3)
    return f_mean, f_std
```

$$\begin{aligned} f_{1:\Delta} &= \phi_e(f_c), \\ \widehat{f}_{1:2^l\Delta} &= (f_{1:2^{l-1}\Delta}, \mathcal{A}^{(l)}(f_{1:2^{l-1}\Delta})), \\ f_p &= (\widehat{f}_{1:\Delta}, \dots, \widehat{f}_{2^{L-1}\Delta:2^L\Delta}). \end{aligned} \tag{3}$$

In each layer, the features from the present f_a (*e.g.*, $f_{1:\Delta}$) are used to approximate the target distribution of the video $p(vid)$. Following [72, 94], in LDA we set the prior distribution of the video as a Gaussian distribution and yield a closed solution of approximation as $p(vid) \sim \mathcal{N}(\mu(f_a) + \mu', \sigma(f_a) + \sigma')$, where μ, σ indicate the mean and variation, and μ', σ' represent the extrapolated ones, respectively. With this simplification, the adaptor can be implemented by a few lines of code: see `est` in Alg. 1. Further, the features from the future f_b can be sampled from the estimated distribution conditioned on the feature from inference layer ϕ_d. As shown in Fig. 4 (a), the future feature can be yielded as

$$f_b = \mathcal{A}(f_a) = (\sigma(f_a) + \sigma')\phi_d(\frac{f_a - \mu(f_a)}{\sigma(f_a)}) + \mu(f_a) + \mu'. \tag{4}$$

Note that f_b has the same length as f_a.

• Spatiotemporal Window U-Net Following [26], we introduce a 3D U-Net $\epsilon_\theta(\boldsymbol{m}_p^t, c)$ parameterized by θ as denoiser. Our spatiotemporal window (STW) U-Net consists of various STW blocks and has the same upsample-downsample architecture structure as usual. Guided by the extrapolated features yield from LDA, STW U-Net takes noised motion cues as input and refines it iteratively. However, it is challenging to efficiently conduct feature interaction between guidance and noise features due to the expensive nature of conventional 3D attention mechanisms. To address this, we propose utilizing a spatiotemporal window attention layer to effectively exploit the feature interactions among them. In detail, we align and fuse the features using a sparse cross-attention mechanism.

To reduce expenditures introduced by attention, STW attention conducts sparse strided attention along with spa-

Table 1. **Ablation Study on KTH** ($u = 10, v = 40$)**.** The lines with blue shallow indicate the optimal setting for our method. If not otherwise specified, this setting is used for all subsequent experiments. CS = Compressed Space. STW = Spatiotemporal-Window Attention. LDA = Layered Distribution Adaptor. OOM = Out-of-Memory.

(a) **Module Ablation**

CS	STW	LDA	SSIM↑	PSNR↑	LPIPS↓	FVD↓
			0.632	23.43	0.182	422.0
✓			0.749	25.39	0.116	307.9
✓	✓		0.778	26.65	0.109	246.2
✓		✓	0.771	27.12	0.103	243.8
✓	✓	✓	**0.799**	**27.91**	**0.093**	**221.4**

(b) **Reconstruction of Autoencoder** ($v = 10$)

Dataset	SSIM↑	PSNR↑	LPIPS↓	FVD↓	FPS↑
BAIR	0.951	28.90	0.014	46.1	183.2
Cityscapes	0.858	27.80	0.048	67.0	146.0
KTH	0.911	33.73	0.027	119.7	182.6
SMMNIST	0.986	32.09	0.008	4.8	182.8
UCF	0.890	28.73	0.030	169.9	163.0

(c) **Operation in LDA**

Variants	PSNR↑	FVD↓
Concat	27.22	249.5
AdaIN	26.45	289.4
AdaIN-z	26.83	263.1
Ours	**27.91**	**221.4**

(d) **Window Size**

Size	PSNR↑	FVD↓
2	26.98	253.6
4	**27.91**	**221.4**
6	27.24	269.1
8	OOM	OOM

(e) **Attention in STW**

Type	PSNR↑	FVD↓
-	27.12	243.8
S-att	27.34	243.7
T-att	27.51	238.9
Ours	**27.91**	**221.4**

(f) **Space Type of CS (PSNR↑)**

Compressed Space	KTH	BAIR	City.	SMM.
Pixel [62]	26.40	17.70	21.90	17.07
Latent [25]	23.43	13.98	14.58	13.38
Flow & occ. map	27.95	18.83	24.34	18.85
Flow & occ. map (GT)	32.11	26.64	23.05	32.14

Table 2. **Architecture Configurations of ExtDM from BAIR.**

Model	ExtDM-K1	ExtDM-K2	ExtDM-K3	ExtDM-K4
# LDA layer	1	2	3	4
# Rate (Pred./Cond.)	1	2	4	8
# Base channels	256	256	256	256
# Channel multiplier	[1,2,4,8]	[1,2,4,8]	[1,2,4,8]	[1,2,4,8]
# STW [t,h,w]	[2,4,4]	[2,4,4]	[2,4,4]	[2,4,4]
Frame resolution	32	32	32	32

tiotemporal dimension in each window, which is shown in Fig. 4 (b). For the guidance f_g from LDA and the feature to be refined f_x, we first split the spatiotemporal features into partitions with a window size of k_w and shift the partition window before the next STW attention similar to [40]. Then, we exploit the spatiotemporal coherence interaction via jointly conducting strided and grid window $\mathcal{T}(\cdot)$. As a result, we estimate the cross-attention as

$$f_{x\to g} = \mathrm{softmax}(\frac{[\mathcal{T}(f_x)\mathbf{W}^{\mathbf{Q}}][\mathcal{T}(f_g)\mathbf{W}^{\mathbf{K}}]^{\top}}{\sqrt{d}})\mathcal{T}(f_x)\mathbf{W}^{\mathbf{V}}. \quad (5)$$

Here, $\mathbf{W}^{\mathbf{Q}}, \mathbf{W}^{\mathbf{K}}, \mathbf{W}^{\mathbf{V}}$ are learnable matrix for linear projection, and d is set as the channel number of features.

Based on STW attention, we construct STW blocks used in our U-Net. With the extracted features from stacked blocks, STW U-Net finally estimates the noises by two separated 3×3 convolution layers that take charge of the occlusion map and flow, respectively. The noises are used for predicting the motion cues $\hat{\boldsymbol{m}}_p$.

4. Experiment

Settings. Following [24, 62], we conduct experiments for short-term and long-term video prediction on five datasets, including KTH [51], BAIR [17], Cityscapes [13], SMMNIST [15, 54], and UCF-101 [53]. We train ExtDM by two stages: (a) perceptual loss [32] for autoencoder and (b) L2 loss [8] for diffusion model. For each ExtDM, we provide architecture configs corresponding to different numbers of LDA as shown in Tab. 2.

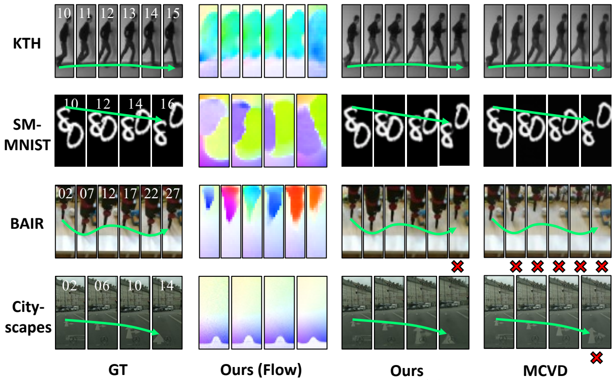


Figure 5. **Qualitative comparison among SOTA methods.** The trajectory of each target is indicated by the green curve.

Metrics. Following [62,77], we employ PSNR, SSIM [67], LPIPS [82] as well as FVD [59] to evaluate the quality of generated videos. Besides, we also evaluate the efficiency of methods, where we report the runtime speed of FPS.

4.1. Ablation Study

To investigate the effectiveness of motion autoencoder, layered distribution adaptor, and spatiotemporal-window attention, we conduct component-wise ablation study in Tab. 1(a). We further discuss each component by conducting in-depth analyses of their variants to answer the following research questions. **RQ1:** Which prediction space performs better? **RQ2:** How to extrapolate features into the future? **RQ3:** What size window is better in STW? **RQ4:** What attention can fuse the extrapolated feature?

RQ1: The latent space has recently attracted the interest of researchers since it amortizes the video as a compact low-dimensional representation for cutting down the computation cost. We conduct experiments on short- and long-term video prediction datasets as shown in Tab. 1 (b)&(f). As can be seen, our method exhibits excellent reconstruction quality, as evidenced by its high PSNR compared to

Table 3. **Quantitative comparisons on KTH** (64 × 64)**.** We compare our method with ten SOTA methods under two settings. **Bold** and underline indicate the highest and second-highest performance. K represents the number of layers in LDA, respectively.

Methods	Year	$u = 10, v = 30$				$u = 10, v = 40$				FPS↑
		SSIM↑	PSNR↑	LPIPS↓	FVD↓	SSIM↑	PSNR↑	LPIPS↓	FVD↓	
U-ViT [6]	CVPR23	0.642	26.13	0.155	694.7	0.606	25.40	0.179	772.0	4.08
DiT [49]	ICCV23	0.657	24.29	0.124	750.4	0.641	23.56	0.145	712.8	2.22
RaMViD [28]	TMLR24	0.590	23.42	0.169	581.3	0.567	22.96	0.187	571.6	0.12
LVDM [25]	ArXiv23	0.644	23.83	0.167	481.1	0.632	23.43	0.182	422.0	1.77
RVD [73]	ArXiv22	0.782	25.32	0.128	441.1	0.758	24.45	0.152	419.1	0.23
VIDM [45]	AAAI23	0.694	25.02	0.150	357.1	0.661	24.32	0.172	376.0	0.89
LFDM [46]	CVPR23	0.772	26.89	0.110	320.2	0.750	26.41	0.116	287.9	3.39
MCVD-c [62]	NeurIPS22	0.812	27.45	0.108	299.8	0.793	26.20	0.124	276.6	6.35
MCVD-cpf [62]	NeurIPS22	0.746	24.30	0.143	294.9	0.720	23.48	0.173	368.4	6.38
MCVD-s [62]	NeurIPS22	0.835	27.50	0.092	323.0	0.744	26.40	0.115	331.6	2.29
ExtDM-K1	CVPR24	0.804	28.34	0.077	284.9	0.784	27.89	0.090	288.7	20.67
ExtDM-K2	CVPR24	0.801	28.43	0.076	239.8	0.779	27.73	0.089	244.1	24.76
ExtDM-K3	CVPR24	0.817	**29.04**	**0.071**	238.3	0.787	**28.31**	**0.084**	231.0	38.36
ExtDM-K4	CVPR24	**0.838**	28.53	0.082	**227.9**	**0.799**	27.91	0.093	**221.4**	**45.28**

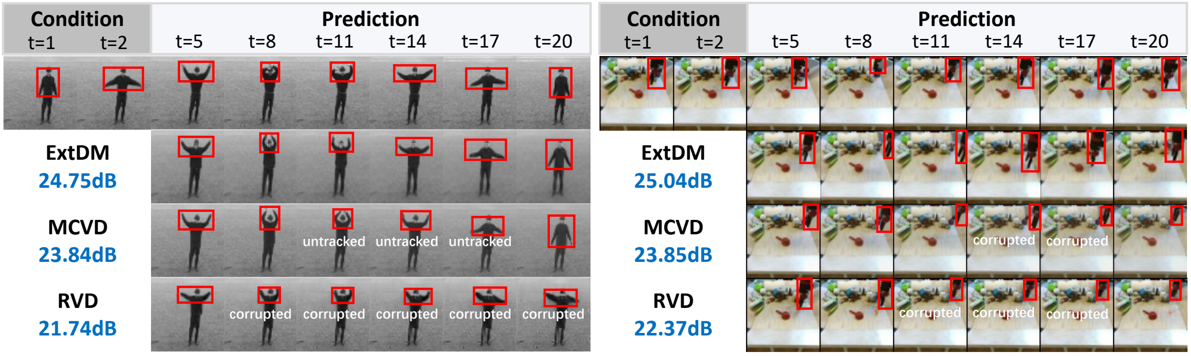


Figure 6. **Qualitative comparison on KTH Action (left) and BAIR (right).** The targets (human arms and the robot arm) are indicated by red boxes. Corresponding PSNRs are illustrated below each video. For more results please refer to Tab. 3 and Tab. 4.

other prediction spaces. We believe that this improvement is attributed to the temporal consistency captured by motion cues, where ExtDM effectively integrates into the prediction along with appearance information. Additionally, we can further enhance the reconstruction quality by incorporating precise flows and occlusion maps from the ground truth data (grey line).

RQ2: Bridging the gap between the future and present is the key point for video prediction. We conduct experiments to verify the effectiveness of our proposed LDA and further discuss its variant in Tab. 1 (a)&(c). As can be seen, our adaptor gains a 9.24% performance boost in ablation and outperforms second-best operations with a margin of 6.89%. We speculate the improvement is because a deterministic trajectory extrapolated by LDA can help the model avoid uncertainty and generate confident predictions. As shown in Fig. 5, our method can predict the following videos with correct trajectories. In contrast, other methods directly anticipate the video based on appearance yielding blurry frames towards the end.

RQ3: As the size of the spatiotemporal window decreases, it is more and more efficient to conduct feature interactions between guidance and plain features. Thus, we explore studying the effect of the window size for our proposed method, as shown in Tab. 1 (a)&(d). As a result, we can conclude that our approach successfully strikes a balance between effectiveness and efficiency when the window size is set to four. The reasons are double-sided. Firstly, a proper window size allows us to control the computational cost while still achieving comparable performance in filtering out irrelevant features, particularly in cases of occlusion. On the other hand, a larger window size can lead to excessive computation and may not capture the relevant information optimally, resulting in subpar performance.

RQ4: Fusing the extrapolated feature can largely encourage the feature interaction between present and future. We conduct experiments in Tab. 1 (a)&(e) to verify the effectiveness of STW Attention. The conclusion we can draw is that the feature fusing requires not only the temporal dimension but also the spatial one for fully exploiting the guidance.

4.2. Comparison with SOTAs

As shown in Tab. 3, Tab. 4, Tab. 5, Tab. 6, and Tab. 7, we demonstrate the main results on two short-term video dataset (*i.e.*, SMMNIST and UCF-101) and three long-term video datasets (*i.e.*, KTH, BAIR, and Cityscapes).

Table 4. **Quantitative comparisons on BAIR** (64 × 64)**.** We compare our method with twelve SOTA methods under two settings.

Methods	Year	$u = 2, v = 14$				$u = 2, v = 28$				FPS↑
		SSIM↑	PSNR↑	LPIPS↓	FVD↓	SSIM↑	PSNR↑	LPIPS↓	FVD↓	
DiT [49]	ICCV23	0.543	15.07	0.171	1013.6	0.520	14.65	0.186	2290.7	2.25
LVDM [25]	ArXiv23	0.464	14.40	0.182	900.6	0.435	13.98	0.198	1663.8	1.31
U-ViT [6]	CVPR23	0.740	17.32	0.078	200.2	0.696	16.53	0.094	263.5	3.84
LFDM [46]	CVPR23	0.770	17.45	0.084	167.6	0.730	16.68	0.106	276.8	5.66
RVD [73]	ArXiv22	0.792	17.88	0.072	139.7	0.750	16.76	0.093	267.1	0.23
RaMViD [28]	TMLR24	0.758	17.55	0.085	166.5	0.691	16.51	0.109	238.7	0.41
VIDM [45]	AAAI23	0.763	16.97	0.080	131.7	0.728	16.20	0.096	194.6	0.82
MCVD-c [62]	NeurIPS22	0.834	19.10	0.078	90.5	0.785	17.60	0.100	120.6	4.15
MCVD-cp [62]	NeurIPS22	0.838	19.10	0.075	87.8	0.797	17.70	0.078	119.0	4.15
MCVD-cpf [62]	NeurIPS22	0.787	17.10	0.077	89.6	0.745	16.20	0.086	118.4	4.15
MCVD-s [62]	NeurIPS22	0.836	19.10	0.078	94.1	0.779	17.50	0.108	132.1	2.51
MCVD-sp [62]	NeurIPS22	0.837	19.20	0.076	90.5	0.789	17.70	0.097	127.9	2.51
ExtDM-K1	CVPR24	0.785	17.73	0.077	114.2	0.748	17.04	0.096	140.3	29.32
ExtDM-K2	CVPR24	0.827	19.76	0.078	97.1	0.790	18.53	0.073	125.8	35.31
ExtDM-K3	CVPR24	0.838	**20.18**	0.066	86.1	0.802	**18.83**	**0.069**	114.7	**37.44**
ExtDM-K4	CVPR24	**0.845**	20.04	**0.053**	**81.6**	**0.814**	18.74	**0.069**	**102.8**	**47.01**

Table 5. **Quantitative comparisons on Cityscapes** (128 × 128)**.**

Methods	Year	$u = 2, v = 28$				FPS↑
		SSIM↑	PSNR↑	LPIPS↓	FVD↓	
U-ViT [6]	CVPR23	0.362	10.84	0.431	1045.3	0.40
RaMViD [28]	TMLR24	0.454	13.14	0.395	812.6	0.12
VIDM [45]	AAAI23	0.539	18.49	0.252	724.7	0.54
RVD [73]	ArXiv22	0.489	17.21	0.242	465.0	0.15
LFDM [46]	CVPR23	0.579	20.32	0.157	194.9	2.93
MCVD-c [62]	NeurIPS22	0.690	21.90	0.112	141.3	2.26
MCVD-s [62]	NeurIPS22	0.720	22.50	0.121	184.8	0.89
ExtDM-K1	CVPR24	0.631	21.49	0.145	157.2	24.65
ExtDM-K2	CVPR24	0.683	21.72	0.135	152.8	28.60
ExtDM-K3	CVPR24	0.701	22.42	0.126	137.2	30.46
ExtDM-K4	CVPR24	**0.745**	**22.84**	**0.108**	**121.3**	**35.44**

Table 6. **Quantitative comparisons on SMMNIST** (64 × 64)**.**

Methods	Year	$u = 10, v = 10$				FPS↑
		SSIM↑	PSNR↑	LPIPS↓	FVD↓	
U-ViT [6]	CVPR23	0.510	17.44	0.138	251.5	4.08
RaMViD [28]	TMLR24	0.585	18.30	0.123	100.4	0.12
LVDM [25]	ArXiv23	0.624	13.38	0.198	49.85	1.77
VIDM [45]	AAAI23	0.514	12.06	0.241	49.08	0.86
LFDM [46]	CVPR23	0.710	15.68	0.137	21.32	3.41
MCVD-c [62]	NeurIPS22	0.786	17.22	0.117	25.63	6.99
MCVD-cpf [62]	NeurIPS22	0.753	16.33	0.139	20.77	6.92
MCVD-s [62]	NeurIPS22	0.785	17.07	0.129	23.86	4.15
MCVD-sf [62]	NeurIPS22	0.758	16.31	0.141	44.14	4.09
MCVD-spf [62]	NeurIPS22	0.748	16.15	0.146	36.12	4.10
RVD [73]	ArXiv22	0.764	18.56	0.123	17.84	0.23
ExtDM-K1	CVPR24	0.776	17.55	0.085	13.87	20.16
ExtDM-K4	CVPR24	**0.813**	**19.59**	**0.068**	**11.11**	**24.54**

Table 7. **Quantitative comparisons on UCF-101** (64 × 64)**.**

Methods	Year	$u = 4, v = 12$				FPS↑
		SSIM↑	PSNR↑	LPIPS↓	FVD↓	
RaMViD [28]	TMLR24	0.639	21.37	0.090	396.7	0.33
LFDM [46]	CVPR23	0.627	20.92	0.098	698.2	3.53
MCVD-cp [62]	NeurIPS22	0.658	21.82	0.088	468.1	1.72
ExtDM-K2	CVPR24	**0.754**	**23.89**	**0.056**	**394.1**	**39.80**

Short-term Video Prediction. We can clearly see the advantage of our network over the alternative (RVD) with a margin of 23.60%. Splitting the motion and appearance of video into the flow autoencoder as well as the diffusion model allows us to identify key motion and content features, and yield predictions with high fidelity compared to others. Therefore, our method avoids producing samples with factual errors, such as the time-varied digits.

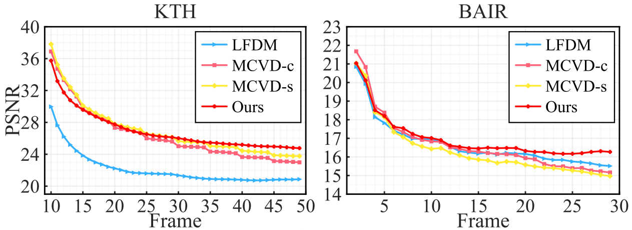


Figure 7. **Frame-wise comparison on long-term video datasets.**

Long-term Video Prediction. First, we can observe a performance improvement of 10.23% among three benchmarks on twenty metrics, which verifies the effectiveness of ExtDM in comparison with second-best metrics from advanced SOTAs (*e.g.*, MCVD, RVD). Second, under two prediction settings on KTH and BAIR, we find the performance degradation for predicting long video (in contrast to the setting of predicting short video) is -9.12%, which is better than -14.60% for the advanced SOTA method (MCVD). Third, we notice that the utilization of LDA can largely increase performance. Compared to the single-layered LDA setting (K1), the multi-layered ones gain an average improvement of 4.73%, 19.41%, 18.17%, and 14.07% on KTH, BAIR, Cityscapes, and SMMNIST, respectively. This is because the temporal distribution is hard to estimate in a time, where we decouple it into multiple steps can yield better results. Besides, introducing multi-layer increases the computation cost, leading to slower processing speed. Thus, ExtDM offers scalable variants to meet the requirements in different scenarios, whether it be for precise performance or high-speed execution.

Per-frame Comparison. To better exploit the temporal consistency, we plot the averaged PSNR over testing videos, alongside the index of video frames. As shown in Fig. 7, we calculate the performance degradation between the first frame and the last one. ExtDM shows low degradation for long-term video prediction, giving us 29.60% better predictions than MCVD (-7.87 v.s -10.20).

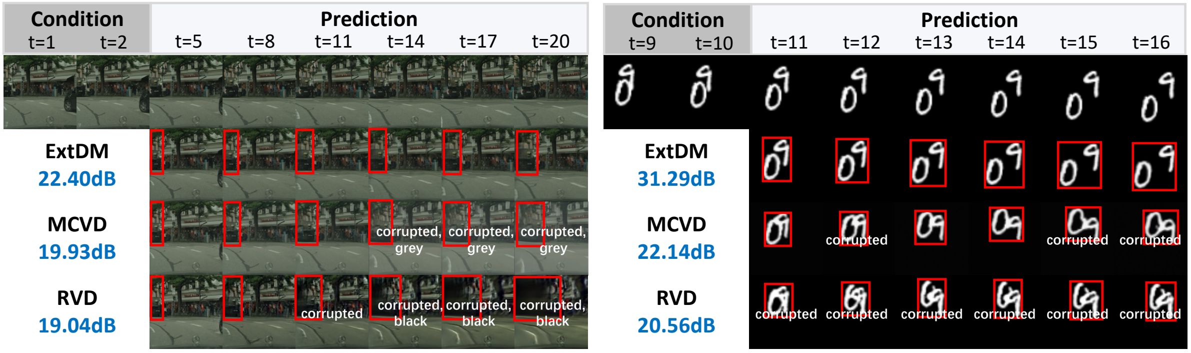
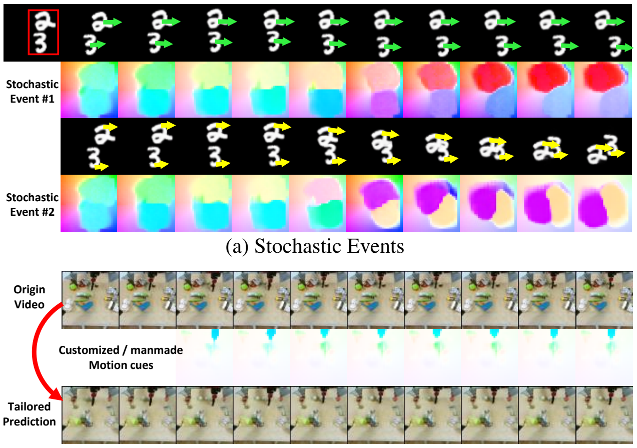


Figure 8. **Qualitative comparison on Cityscapes and Stochastic Moving MNIST.** The targets (the car on the left and two digits on the right) are indicated by red boxes. Corresponding PSNRs are illustrated below each video. For more results please refer to Tab. 5 and Tab. 6.

Visualization Comparison. Fig. 6 and Fig. 8 show the qualitative comparison between ExtDM and SOTA methods. We find that ExtDM can generate temporal consistent results and avoids factual errors (*e.g.*, incorrect motion, gradual grey, corrupted human, digits, and background) in existing SOTA methods. This again verifies our insight that existing methods mostly focus on the cues from the present due to overlooking the dynamic changes in the future.

Runtime Analysis. Experimental results on five datasets verify the efficiency of our method, where ours runs 7.51 times faster. The improvement comes from avoiding the heavy cost involved in computing high-resolution frames and instead estimating low-resolution motion cues.



(a) Stochastic Events

(b) Tailored Predictions

Figure 9. Prediction results for (a) stochastic events on SMMNIST and (b) tailored prediction on BAIR.

5. Discussion

ExtDM-based video generation framework can boost various directions. Here, we envision two potential uses.

Stochastic Events. For a considerable time, predicting stochastic events has been an aspirational objective in the field, and with the aid of ExtDM, it is now feasible to take an additional leap forward. If stochastic events follow the physical disciplines such as changing directions and velocities when reaching the boundary [15], we can design some rules to generate motion cues accordingly. Then, ExtDM can naturally generate potential predictions according to the principled motion cues, as shown in Fig. 9 (a).

Tailored Perdiction. In addition to generating tailored predictions, a man can customize a preferred trajectory ideally. Based on ExtDM, we decouple the video prediction as motion cues extrapolation and video frame reconstruction, thus making that modify the prediction results according to the human-made motion cues. Fig. 9 (b) shows a customized prediction by extracting motion cues from another video.

6. Conclusion

In this paper, we propose ExtDM, a diffusion-based framework for video prediction by extrapolating the distribution along temporal dimensions. We reformulate the video prediction task as a problem of estimating the future cues of the video, thus providing a solution that extrapolates the distribution shift from present to future. The extrapolation mainly focuses on i) modeling temporal dynamics in both the short- and long-term future, and ii) constructing video distribution along with temporal dimension in a pairwise manner. As a low-hanging fruit, the extrapolated cues can be efficiently predicted due to their compacted resolution and be customized by humans to multiple potential proposals of the future. Overall, ExtDM helps the video prediction more effective and can be interpreted from the viewpoint of intermediate future cues. Extensive experiments on fivevideo prediction datasets show our model achieves a new SOTA for both short- and long-term video prediction.

Acknowledgments. This work was supported by the Natural Science Foundation of Tianjin, China (NO.20JCJQJC00020), the National Natural Science Foundation of China (No.62306154), Fundamental Research Funds for the Central Universities, Supercomputing Center of Nankai University (NKSC), and in part by the Ministry of Education and Science of Bulgaria for INSAIT.

References

[1] Merin Abraham, Nikita Suryawanshi, Nevin Joseph, and Dhanashree Hadsul. Future predicting intelligent camera security system. In *ICIT*, 2021. 1

[2] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Towards high resolution video generation with progressive growing of sliced wasserstein gans. *arXiv*, 2018. 1

[3] Adil Kaan Akan, Erkut Erdem, Aykut Erdem, and Fatma Güney. Slamp: Stochastic latent appearance and motion prediction. In *ICCV*, 2021. 2

[4] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. In *ICLR*, 2017. 2

[5] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H. Campbell, and Sergey Levine. Stochastic variational video prediction. In *ICLR*, 2018. 1

[6] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *CVPR*, 2023. 6, 7

[7] Apratim Bhattacharyya, Mario Fritz, and Bernt Schiele. Long-term on-board prediction of people in traffic scenes under uncertainty. In *CVPR*, 2018. 1

[8] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 2, 5

[9] Lluis Castrejon, Nicolas Ballas, and Aaron Courville. Improved conditional vrnns for video prediction. In *ICCV*, 2019. 2

[10] Zheng Chang, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. Strpm: A spatiotemporal residual predictive model for high-resolution video prediction. In *CVPR*, 2022. 2

[11] Xiuwen Chen, Li Fang, Long Ye, and Qin Zhang. Deep video harmonization by improving spatial-temporal consistency. *MIR*, 21(1):46–54, 2024. 2

[12] Yadang Chen, Chuanyan Hao, Zhi-Xin Yang, and Enhua Wu. Fast target-aware learning for few-shot video object segmentation. *SCIS*, 65(8):182104, 2022. 2

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 5

[14] Aram Davtyan, Sepehr Sameni, and Paolo Favaro. Randomized conditional flow matching for video prediction. *arXiv*, 2022. 1, 2

[15] Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *ICML*, 2018. 2, 5, 8

[16] Frederik Ebert, Chelsea Finn, Sudeep Dasari, Annie Xie, Alex Lee, and Sergey Levine. Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. *arXiv*, 2018. 1

[17] Frederik Ebert, Chelsea Finn, Alex X. Lee, and Sergey Levine. Self-supervised visual planning with temporal skip connections. In *CoRL*, 2017. 5

[18] Jean-Yves Franceschi, Edouard Delasalles, Mickaël Chen, Sylvain Lamprier, and Patrick Gallinari. Stochastic latent residual video prediction. In *ICML*, 2020. 1

[19] Tsu-Jui Fu, Xin Eric Wang, Scott T. Grafton, Miguel P. Eckstein, and William Yang Wang. M3l: Language-based video editing via multi-modal multi-level transformers. In *CVPR*, 2022. 1

[20] Antonino Furnari and Giovanni Maria Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *ICCV*, 2019. 1

[21] Naoya Fushishita, Antonio Tejero-de Pablos, Yusuke Mukuta, and Tatsuya Harada. Long-term human video generation of multiple futures using poses. In *ECCV*, 2020. 1

[22] Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z. Li. Simvp: Simpler yet better video prediction. In *CVPR*, 2022. 1

[23] Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *CVPR*, 2020. 2

[24] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. In *NeurIPS*, 2022. 1, 5

[25] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv*, 2023. 2, 5, 6, 7

[26] Jonathan Ho, Tim Salimans, Alexey A. Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *NeurIPS*, 2022. 1, 2, 4

[27] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *JMLR*, 14(40):1303–1347, 2013. 1

[28] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *TMLR*, 2024. 2, 6, 7

[29] Duojun Huang, Xinyu Xiong, De-Jun Fan, Feng Gao, Xiao-Jian Wu, and Guanbin Li. Annotation-efficient polyp segmentation via active learning. *arXiv*, 2024. 2

[30] Duojun Huang, Xinyu Xiong, Jie Ma, Jichang Li, Zequn Jie, Lin Ma, and Guanbin Li. Alignsam: Aligning segment anything model to open context via reinforcement learning. In *CVPR*, 2024. 2

[31] Guoli Jia and Jufeng Yang. S 2-ver: Semi-supervised visual emotion recognition. In *ECCV*, 2022. 2

[32] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 3, 5

[33] Hema S Koppula and Ashutosh Saxena. Anticipating human activities using object affordances for reactive robotic response. *TPAMI*, 38(1):14–29, 2015. 1

[34] Yong-Hoon Kwon and Min-Gyu Park. Predicting future frames using retrospective cycle gan. In *CVPR*, 2019. 2

[35] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 2

[36] Alex X Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv*, 2018. 1, 2

[37] Jiaming Li, Jiacheng Zhang, Jichang Li, Ge Li, Si Liu, Liang Lin, and Guanbin Li. Learning background prompts to discover implicit knowledge for open vocabulary object detection. In *CVPR*, 2024. 2

[38] Xin Liu, Guobao Xiao, Riqing Chen, and Jiayi Ma. Pgfnet: Preference-guided filtering network for two-view correspondence learning. *TIP*, 32:1367–1378, 2023. 2

[39] Xin Liu and Jufeng Yang. Progressive neighbor consistency mining for correspondence pruning. In *CVPR*, 2023. 2

[40] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 5

[41] Haoyu Lu, Guoxing Yang, Nanyi Fei, Yuqi Huo, Zhiwu Lu, Ping Luo, and Mingyu Ding. Vdt: An empirical study on video diffusion with transformers. *arXiv*, 2023. 2

[42] Siwei Ma, Junlong Gao, Ruofan Wang, Jianhui Chang, Qi Mao, Zhimeng Huang, and Chuanmin Jia. Overview of intelligent video coding: from model-based to learning-based approaches. *VI*, 1(1):15, 2023. 1

[43] Siwei Ma, Li Zhang, Shiqi Wang, Chuanmin Jia, Shanshe Wang, Tiejun Huang, Feng Wu, and Wen Gao. Evolution of avs video coding standards: twenty years of innovation and development. *SCIS*, 65(9):192101, 2022. 2

[44] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2017. 2

[45] Kangfu Mei and Vishal M Patel. Vidm: Video implicit diffusion models. *AAAI*, 2023. 6, 7

[46] Haomiao Ni, Changhao Shi, Kai Li, Sharon X. Huang, and Martin Renqiang Min. Conditional image-to-video generation with latent flow diffusion models. In *CVPR*, 2023. 6, 7

[47] Yaniv Nikankin, Niv Haim, and Michal Irani. Sinfusion: Training diffusion models on a single image or video. In *ICML*, 2023. 2

[48] Sergiu Oprea, Pablo Martinez-Gonzalez, Alberto Garcia-Garcia, John Alejandro Castro-Vargas, Sergio Orts-Escolano, Jose Garcia-Rodriguez, and Antonis Argyros. A review on deep learning techniques for video prediction. *TPAMI*, 44(6):2806–2826, 2020. 1

[49] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 6, 7

[50] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1

[51] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *ICPR*, 2004. 5

[52] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *NeurIPS*, 2015. 2

[53] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv*, 2012. 5

[54] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *ICML*, 2015. 5

[55] Mingzhen Sun, Weining Wang, Xinxin Zhu, and Jing Liu. Moso: Decomposing motion, scene and object for video prediction. *arXiv*, 2023. 1, 2

[56] Bin Tan, Nan Xue, Song Bai, Tianfu Wu, and Gui-Song Xia. Planetr: Structure-guided transformers for 3d plane recovery. In *ICCV*, 2021. 1

[57] Bin Tan, Nan Xue, Tianfu Wu, and Gui-Song Xia. Nopesac: Neural one-plane ransac for sparse-view planar 3d reconstruction. *TPAMI*, 2024. 1

[58] Stephen Tian, Chelsea Finn, and Jiajun Wu. A control-centric benchmark for video prediction. In *ICLR*, 2023. 2

[59] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv*, 2018. 5

[60] Angel Villar-Corrales, Ani Karapetyan, Andreas Boltres, and Sven Behnke. Mspred: Video prediction at multiple spatio-temporal scales with hierarchical recurrent networks. In *BMVC*, 2022. 1

[61] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. In *ICLR*, 2017. 1

[62] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation. In *NeurIPS*, 2022. 1, 2, 5, 6, 7

[63] Lijuan Wang, Guoli Jia, Ning Jiang, Haiying Wu, and Jufeng Yang. Ease: Robust facial expression recognition via emotion ambiguity-sensitive cooperative networks. In *ACM MM*, 2022. 2

[64] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S. Yu. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *ICML*, 2018. 1

[65] Yunbo Wang, Lu Jiang, Ming-Hsuan Yang, Li-Jia Li, Mingsheng Long, and Li Fei-Fei. Eidetic 3d lstm: A model for video prediction and beyond. In *ICLR*, 2018. 2

[66] Yunbo Wang, Haixu Wu, Jianjin Zhang, Zhifeng Gao, Jianmin Wang, Philip Yu, and Mingsheng Long. Predrnn: A recurrent neural network for spatiotemporal predictive learning. *TPAMI*, 45(2):2208–2225, 2022. 1, 2

[67] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 13(4):600–612, 2004. 5

[68] Changsong Wen, Guoli Jia, and Jufeng Yang. Dip: Dual incongruity perceiving network for sarcasm detection. In *CVPR*, 2023. 2

[69] Changsong Wen, Xin Zhang, Xingxu Yao, and Jufeng Yang. Ordinal label distribution learning. In *ICCV*, 2023. 2

[70] Haixu Wu, Zhiyu Yao, Jianmin Wang, and Mingsheng Long. Motionrnn: A flexible model for video prediction with spacetime-varying motions. In *CVPR*, 2021. 2

[71] Jinbo Xing, Wenbo Hu, Yuechen Zhang, and Tien-Tsin Wong. Flow-aware synthesis: A generic motion model for video frame interpolation. *CVMJ*, 7(3):393–405, 2021. 1

[72] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *ECCV*, 2020. 4

[73] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv*, 2022. 2, 6, 7

[74] Siyuan Yang, Lu Zhang, Yu Liu, Zhizhuo Jiang, and You He. Video diffusion models with local-global context guidance. In *IJCAI*, 2023. 2

[75] Xi Ye and Guillaume-Alexandre Bilodeau. Vptr: Efficient transformers for video prediction. *ICPR*, 2022. 2

[76] Xi Ye and Guillaume-Alexandre Bilodeau. A unified model for continuous conditional video prediction. In *CVPR*, 2023. 2

[77] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. *arXiv*, 2022. 5

[78] Sihyun Yu, Kihyuk Sohn, Subin Kim, and Jinwoo Shin. Video probabilistic diffusion models in projected latent space. In *CVPR*, 2023. 2

[79] Wei Yu, Yichao Lu, Steve Easterbrook, and Sanja Fidler. Efficient and information-preserving future frame prediction and beyond. In *ICLR*, 2020. 2

[80] Yingjie Zhai, Guoli Jia, Yu-Kun Lai, Jing Zhang, Jufeng Yang, and Dacheng Tao. Looking into gait for perceiving emotions via bilateral posture and movement graph convolutional networks. *TAC*, 2024. 1

[81] Pengyu Zhang, Dong Wang, and Huchuan Lu. Multi-modal visual tracking: Review and experimental comparison. *CVMJ*, 10(2):193–214, 2024. 1

[82] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[83] Zhicheng Zhang, Song Chen, Zichuan Wang, and Jufeng Yang. Planeseg: Building a plug-in for boosting planar region segmentation. *TNNLS*, 2024. 1

[84] Zhicheng Zhang, Shengzhe Liu, and Jufeng Yang. Multiple planar object tracking. In *ICCV*, 2023. 1

[85] Zhicheng Zhang, Lijuan Wang, and Jufeng Yang. Weakly supervised video emotion detection and prediction via cross-modal temporal erasing network. In *CVPR*, 2023. 1

[86] Zhicheng Zhang and Jufeng Yang. Temporal sentiment localization: Listen and look in untrimmed videos. In *ACM MM*, 2022. 1

[87] Zhicheng Zhang, Pancheng Zhao, Eunil Park, and Jufeng Yang. Mart: Masked affective representation learning via masked temporal distribution distillation. In *CVPR*, 2024. 2

[88] Pancheng Zhao, Peng Xu, Pengda Qin, Deng-Ping Fan, Zhicheng Zhang, Guoli Jia, Bowen Zhou, and Jufeng Yang. Lake-red: Camouflaged images generation by latent background knowledge retrieval-augmented diffusion. In *CVPR*, 2024. 2

[89] Sicheng Zhao, Guoli Jia, Jufeng Yang, Guiguang Ding, and Kurt Keutzer. Emotion recognition from multiple modalities: Fundamentals and methodologies. *SPM*, 38(6):59–73, 2021. 2

[90] Sicheng Zhao, Xingxu Yao, Jufeng Yang, Guoli Jia, Guiguang Ding, Tat-Seng Chua, Björn W. Schuller, and Kurt Keutzer. Affective image content analysis: Two decades review and new perspectives. *TPAMI*, 44(10):6729–6751, 2022. 2

[91] Shihao Zhou, Duosheng Chen, Jinshan Pan, Jinglei Shi, and Jufeng Yang. Adapt or perish: Adaptive sparse transformer with attentive feature refinement for image restoration. In *CVPR*, 2024. 2

[92] Shihao Zhou, Mengxi Jiang, Shanshan Cai, and Yunqi Lei. Dc-gnet: Deep mesh relation capturing graph convolution network for 3d human shape reconstruction. In *ACM MM*, 2021. 2

[93] Shihao Zhou, Mengxi Jiang, Qicong Wang, and Yunqi Lei. Towards locality similarity preserving to 3d human pose estimation. In *ACCV*, 2020. 2

[94] Xiaosu Zhu, Jingkuan Song, Lianli Gao, Feng Zheng, and Heng Tao Shen. Unified multivariate gaussian mixture for efficient neural image compression. In *CVPR*, 2022. 4