

Decouple Content and Motion for Conditional Image-to-Video Generation

Cuifeng Shen¹, Yulu Gan¹, Chen Chen^{2*}, Xiongwei Zhu³, Lele Cheng³, Tingting Gao³, Jinzhi Wang^{1*}

¹Peking University

²Chinese Academy of Science

³Kuaishou Technology

{cuifeng.shen,ganyulu}@stu.pku.edu.cn, chen.chen@ia.ac.cn, {zhuxiongwei, lisize}@kuaishou.com, jinzhiw@pku.edu.cn

Abstract

The goal of conditional image-to-video (cI2V) generation is to create a believable new video by beginning with the condition, i.e., one image and text. The previous cI2V generation methods conventionally perform in RGB pixel space, with limitations in modeling motion consistency and visual continuity. Additionally, the efficiency of generating videos in pixel space is quite low. In this paper, we propose a novel approach to address these challenges by disentangling the target RGB pixels into two distinct components: spatial content and temporal motions. Specifically, we predict temporal motions which include motion vector and residual based on a 3D-UNet diffusion model. By explicitly modeling temporal motions and warping them to the starting image, we improve the temporal consistency of generated videos. This results in a reduction of spatial redundancy, emphasizing temporal details. Our proposed method achieves performance improvements by disentangling content and motion, all without introducing new structural complexities to the model. Extensive experiments on various datasets confirm our approach's superior performance over the majority of state-of-the-art methods in both effectiveness and efficiency.

Introduction

Deep generative models have garnered wide attention, driven by the emergence of approaches like Generative Adversarial Models (Goodfellow et al. 2020) and Diffusion Models (Ho, Jain, and Abbeel 2020). These models have recently demonstrated remarkable success in tasks such as image generation. In the field of the conditional Image-to-Video (cI2V), most works (Ramesh et al. 2021; He et al. 2022a; Yu et al. 2022; Mei and Patel 2022; Voleti, Jolicœur-Martineau, and Pal 2022; Blattmann et al. 2023) have showcased the potential and versatility of these cutting-edge techniques.

A line of works (Yu et al. 2022; Mei and Patel 2022; Voleti, Jolicœur-Martineau, and Pal 2022) in cI2V aims to directly extend images into videos by generating a series of image frames in RGB space (see in Fig 1(b)). However, they encounter the following challenges: (1) *Information redundancy* of videos makes it difficult for a model to focus on the video's important temporal information. The slight variations present

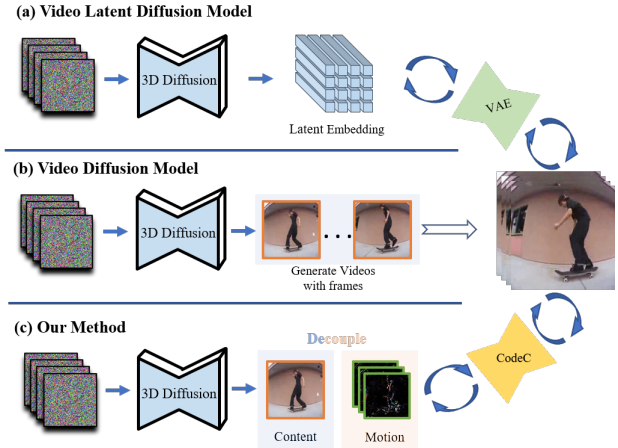


Figure 1: Motivations and our ideas. Conventional methods (see in (b)), involve extending the RGB space with time sequences, resulting in limited memory efficiency and temporal coherence. Latent Diffusion Models employ a variational autoencoder for compression (depicted in (a)), enhancing efficiency but potentially reducing spatial quality and poor temporal coherence because temporal consistency hasn't been directly modeled. Our approach (refer to (c)) decouples the content and motion, capitalizing on existing temporal coherence in compressed video data, resulting in a memory-efficient and temporally consistent video generation approach.

in each frame, combined with inherent redundancies within the video space, lead to a neglect of temporal details for the model when attempting to construct the video. This neglect hinders the ability of the model to focus on sequential video frames effectively. Consequently, the process of pixel-based generation can cause the model to disproportionately highlight the spatial content, thereby complicating the modeling of temporal motions. Achieving accurate and efficient generation of temporal information is notably demanding. (2) *Time consuming*. Generating the whole video for each frame in pixel space consumes significant resources, i.e., for a 16-frame $128 \times 128 \times 3$ video, the target vector dimension would be $16 \times 128 \times 128 \times 3$.

To address the high computational request of directly ex-

*Corresponding author.

tending images into videos, some work (Blattmann et al. 2023) encodes the image into latent space (see in Fig. 1(a)). However, this approach can cause a reduction in the quality of the frames due to the use of VAE, and still can not generate temporal consistent videos. Recent work LFDM (Ni et al. 2023) uses a diffusion model to predict optical flow and then uses the optical flow in conjunction with the original image content to generate videos. However, optical flow cannot be easily inverted and is inaccurate, leading to the use of specialized flow predictors that may encounter local optima.

The approach of separating time and content information holds great potential. First, we can model the temporal information of videos individually, rather than considering all pixels together. Secondly, this approach allows us to save a considerable amount of computational cost. In comparison to LFDM, our method uses rigorous computation to model temporal information and is invertible.

Specifically, we first employ a simpler approach, named *Decouple-Based Video Generation* (D-VDM) to directly predict the differences between two consecutive frames. Then we propose the *Efficient Decouple-Based Video Generation* (ED-VDM) method. We separate the content and temporal information of videos using a CodeC (Le Gall 1991) to extract motion vectors and residual content. During training, we input the motion vectors, residual, and the first frame image together. The predictive model generates the motion vectors and residual output, and then we use the CodeC decoder to warp them with the image to restore the video. As we decouple the temporal and content information, during the prediction process, we aim for the joint probability distribution of input motion vectors and residual, while the model outputs the score of that distribution. Diffusion models have been proven effective in learning the score of joint distribution (Bao et al. 2023).

To recap, our main contributions are as follows:

- Our proposed D-VDM decouples the video into content and temporal motions, enabling explicit modeling of the temporal motions. To model the decouplings separately, we use a diffusion-based method to model the temporal motions of a video and warp it to the given first frame.
- We investigate various compression techniques to decrease the spatial dimension of the temporal motion features. Our proposed ED-VDM, which employs an autoencoder to compress residuals, provides a 110x improvement in training and inference speed while maintaining state-of-the-art (SOTA) performance.
- Extensive experiments are conducted, and the SOTA performance has been achieved on MHAD (Chen, Jafari, and Kehtarnavaz 2015), NATOPS (Yale Song and Davis 2011), and BAIR (Ebert et al. 2017) datasets, demonstrating that our model can generate realistic and temporal consistent video.

Related Work

Diffusion Model

Diffusion denoising probabilistic models (DDPMs) (Sohl-Dickstein et al. 2015) learn to generate data samples through a sequence of denoising autoencoders that estimate the

score (HyvärinenAapo 2005) of the data distribution (a direction pointing toward higher density data). Recently, diffusion probabilistic models (Springenberg 2015; Ho, Jain, and Abbeel 2020; Song et al. 2020; Karras et al. 2022) achieve remarkable progress in image generation (Rombach et al. 2022; Bao et al. 2022), text-to-image generation (Nichol et al. 2023; Ramesh et al. 2023; Gu et al. 2022), 3D scene generation (Poole et al. 2023) and image editing (Meng et al. 2021; Choi et al. 2021).

Our video generation approach leverages the outstanding performance of the diffusion model, with the primary goal of generating the video motion feature, thereby achieving better video temporal consistency. Furthermore, we reduce the spatial redundancy in generation processes by compressing the temporal feature.

Video Generation with Diffusion Model

Video generation aims to generate a sequence of images with temporal consistency. In the context of diffusion-based methods, VDM (Ho et al. 2022b) extended the 2D Unet architecture to 3D, and generate 3D video signals. Make-a-Video (Singer et al. 2022), ImageN-Video (Ho et al. 2022a), and Phenaki (Villegas et al. 2022) have applied similar architecture to generate high-resolution and long-duration videos, resulting in high computational resources. To get rid of high GPU memory usage, MCVD (Voleti, Jolicoeur-Martineau, and Pal 2022) generates videos in a temporal auto-regressive manner to reduce memory usage. PVDM (Yu et al. 2023) and LVDM (He et al. 2022a) propose a latent diffusion model that utilizes a VAE to compress RGB pixels to latent space. However, the RGB pixels usually only contribute a limited $8\times$ spatial downsample ratio. Unlike previous methods that generate videos in RGB space or VAE latent space, our approach transfers the target space into a compressed temporal information space, resulting in significant performance improvements and a notable spatial downsample ratio ($16\times$).

Video Compression

Video compression is to reduce the amount of data required to store or transmit a video by removing redundant information. MPEG-4 (Le Gall 1991) is one of the most commonly used methods for video compression. MPEG-4 uses CodeC to decompose a video into I-frames and P-frames, representing the first frame and temporal information respectively. P-frames encode temporal information via motion vectors and residuals for object movement and image variance.

Computer Vision on Decoupled Videos

Video decoupling based methods means represents a video in spatial and temporal information, and use both information to improve the performance of computer vision task. Recent works, such as (Cheng, Tai, and Tang 2021; Yang and Yang 2022; Huang et al. 2021), have highlighted the importance of video decoupling in video perception and understanding. Moreover, some methods have achieved impressive performance in video understanding by decoupling the video into the first frame and motion vectors and residuals. (Wu et al. 2018) trained a deep network on compressed videos, which

simplifies training and utilizes the motion information already presents in compressed videos. (Huang et al. 2021) use key frames and motion vectors in compressed videos as supervision to improve the performance of video representations.

Decouple Content and Motion Generation

The goal of conditional image-to-video is to generate a video given the first frame and conditions. Assume $\mathbf{s} \sim \mathcal{N}(0, I)$ is a Gaussian noise with the shape of $N \times H \times W \times C$ where N , H , W , and C represents the number of frames, height, width, and channel respectively. Denote x_0 as the first frame of a video clip, and $\mathbf{x}_0 = \{x_0, x_1, \dots, x_K\}$ represents the video clip which has the same shape as the Gaussian noise. During training, the diffusion model learns the score of the video distribution. During sampling, starting with the initial frame x_0 and condition y we generate a video clip $\hat{\mathbf{x}}_0 = \{\hat{x}_0, \hat{x}_1, \dots, \hat{x}_K\}$ from the learned distribution, beginning with Gaussian noise $\mathbf{s}$. Based on the datasets, we only use text labels as the condition y . In this section, we first introduce the preliminary diffusion models and explain our proposed D-VDM and ED-VDM methods in detail.

Diffusion Model

Denoising Diffusion Possibility Model (DDPM) (Ho, Jain, and Abbeel 2020) consists of a forward process that perturbs the data to a standard Gaussian distribution and a reverse process that starts with the given Gaussian distribution and uses a denoising network to gradually restore the undisturbed data structure.

Specifically, consider $\mathbf{x}_0$ as a data sample from the distribution $\mathbf{x}_0 \sim q(\mathbf{x}_0)$, representing a video clip in this study. We denote T as the total step of the perturbation. In the forward process, DDPM produces a Markov chain $\mathbf{x}_0, \dots, \mathbf{x}_T$ by injecting Gaussian noise $\mathcal{N}(0, I)$ to $\mathbf{x}_0$, that is:

$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_{t-1}; \sqrt{\alpha_t} \mathbf{x}_{t-1}, \beta_t I). \quad (1)$$

$$q(\mathbf{x}_{1:T} | \mathbf{x}_0) = \prod_{t=1}^T q(\mathbf{x}_t | \mathbf{x}_{t-1}), \quad (2)$$

where $\alpha_t = 1 - \beta_t$ and β_t is the noise schedule.

Regarding the denoising process, when the value of T is sufficiently large, the posterior distribution $q(\mathbf{x}_{t-1} | \mathbf{x}_t)$ can be approximated as a Gaussian distribution. The reverse conditional probability can be computed using Bayes' rule conditioned on x_0 :

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) := \mathcal{N}(x_t; \hat{\mu}(\mathbf{x}_t, \mathbf{x}_0), \hat{\sigma}), \quad (3)$$

where $\hat{\mu}(\mathbf{x}_t, \mathbf{x}_0)$ is obtained as:

$$\hat{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0, \quad (4)$$

Moreover, by integrating equation 2, predicting the original video $\mathbf{x}_0$ is equivalent to predict the noise ϵ added in $\mathbf{x}_t$:

$$\hat{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_t \right), \quad (5)$$

Hence, to estimate $\hat{\mu}(\mathbf{x}_t, \mathbf{x}_0)$, we need to learn the function $\mu_\theta(\mathbf{x}_t, t)$. We can achieve this by directly learning the noise $\epsilon_\theta(\mathbf{x}_t, t)$:

$$\mathbb{E}_{t \sim \mathcal{U}(0, T), \mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(0, 1)} [\lambda(t) \|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2], \quad (6)$$

Decoupled Video Diffusion Model

Video diffusion models aim to use DDPM to estimate the score of the video distribution $\mathbf{v}_0 \sim q(\mathbf{v}_0)$, where $\mathbf{v}_0 = \{v_0, v_1, \dots, v_K\}$ belongs to the RGB pixel space $\mathbb{Z}_{[0, 255]}^{3 \times K \times W \times H}$, and K is the frame number, W and H is frame width and height respectively. The denoising 3D Unet is designed to learn a denoising parameter $\epsilon_\theta(\mathbf{v}_t, t)$.

One simple approach to decouple a video into spatial and temporal representations, as illustrated in Figure 3, is to retain the first frame and then compute the differences between it and the subsequent frames, noted as $\hat{\mathbf{v}}_0 \in \mathcal{V}(\hat{\mathbf{v}}_0)$, where $\hat{\mathbf{v}}_0^n = \mathbf{v}_0^n - \mathbf{v}_0^{n-1}$, $\hat{v}_0^{2 \dots n} \sim \mathbb{Z}_{[-255, 255]}^{3 \times K-1 \times W \times H}$.

To align the difference with the first frame and provide spatial information, low-level semantic information of the first frame is incorporated into the learning target. This is achieved by using a ResNet (He et al. 2016) bottleneck module to encode the first frame (denoted as $\tau_\theta(v_0)$) and concatenating it to the learning target along the channel dimension. Subsequently, the learning objective is slightly modified as

$$L = \mathbb{E}_{t, \hat{\mathbf{v}}_0 \sim \mathcal{V}(\hat{\mathbf{v}}_0), \epsilon \sim \mathcal{N}(0, 1)} [\lambda(t) \|\epsilon - \epsilon_\theta(\hat{\mathbf{v}}_t, t, \tau_\theta(v_0))\|^2], \quad (7)$$

where τ_θ and ϵ_θ is jointly optimized.

Efficient Decoupled Video Diffusion Model

As discussed in the previous section, another efficient representation of decoupled video can be achieved through I-frames and P-frames. P-frames comprise motion vectors and residuals, as depicted in Figure 3.

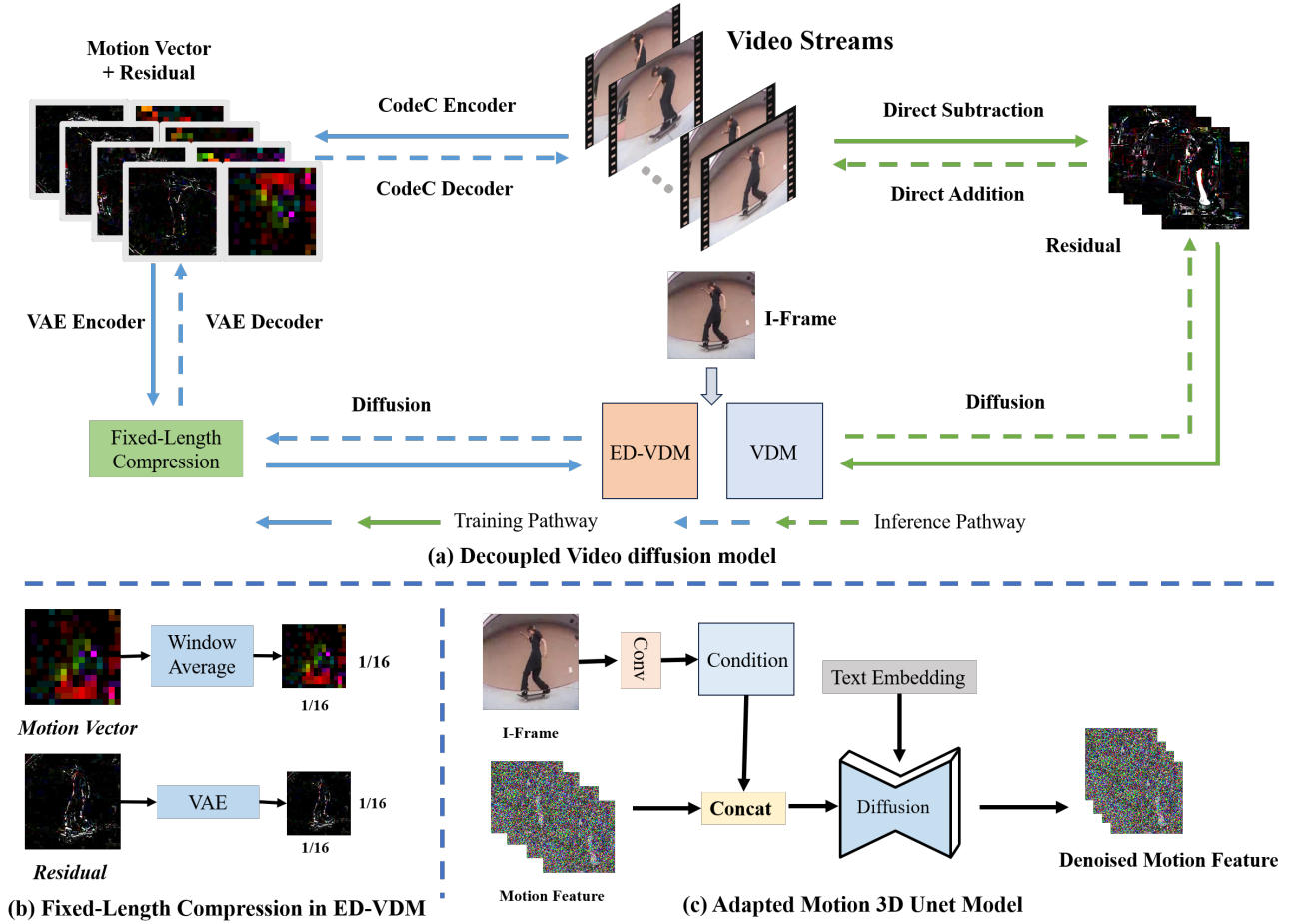
We follow the H.264 protocol to obtain the motion vector $\mathbf{m}$ and residuals $\mathbf{r}$ from a video tube $\mathbf{v}$, using a reversible function $f(\mathbf{v}) = \langle \mathbf{m}, \mathbf{r} \rangle$:

Let $\mathbf{v}^n$ and $\mathbf{v}^{n-1}$ be the current and the previous frame, respectively. We divide $\mathbf{v}^{n-1}$ into non-overlapping micro-blocks of size 16×16 pixels, denoted as B_i , where i represents the index of the micro-blocks. To obtain the motion vector $\mathbf{m}_i$ for each micro-blocks B_i , we search for a corresponding block B'_i in the current frame $\mathbf{v}^n$ that is similar to B_i . This can be achieved by minimizing the sum of absolute differences between the two blocks, which can be formulated as:

$$\mathbf{m}_i = \underset{\mathbf{u}, \mathbf{w}}{\operatorname{argmin}} \sum_{j, k} |B_i(j, k) - B'_i(j + \mathbf{u}, k + \mathbf{w})|$$

Once the motion vector $\mathbf{m}_i$ is obtained, the residual $\mathbf{r}$ can be calculated as the difference between the previous micro-blocks B_i and the motion-compensated block B'_i in the current frame, i.e., $\mathbf{r}_i = B_i - B'_i(\mathbf{m}_i)$. The motion vectors and residuals for all micro-blocks can then be combined to form the P-frame.

Motion vectors that are used to represent the motion information of 16×16 blocks in the video frames contain



A

Figure 2: Illustration of our proposed decoupled video diffusion model. (a) Pipeline. The green pathway represents the Decoupled Video Diffusion Model (D-VDM), which directly generates motion features in the compressed video domain, while the blue pathway illustrates the Efficient Decoupled Video Diffusion Model (ED-VDM), which includes a reversible compression function. (b) Compression techniques used in the ED-VDM model. Since the separated motion vectors and residuals are of unequal lengths, it is necessary for us to apply equal-length processing to both components. (c) The architecture of the 3D U-Net. We employed the 3D U-Net architecture in both models.

identical numbers, and thus can achieve a high spatial compression ratio of $256\times$. Residuals, on the other hand, have the same spatial size as the video frames, but contain less information than the original frames and thus can be compressed efficiently. For the residual compression, we utilize a Latent Diffusion (Rombach et al. 2022) autoencoder to compress the residuals into a latent space. In order to match the spacial dimension of the motion vector, the residual down-sampling rate is set equal to the motion vector compression rate and in our case is $16\times$. Specifically, given a residual $\mathbf{r} \in \mathbb{Z}_{[-255, 255]}^{3 \times W \times H}$, the encoder $\mathcal{E}$ encodes the residual to a latent space $\mathbf{z} = \mathcal{E}(\mathbf{r}) \in \mathbb{R}^{16 \times \frac{W}{16} \times \frac{H}{16}}$, and the decoder $\mathcal{D}$ could reconstruct the image from the latent $\mathbf{r}' = \mathcal{D}(\mathcal{E}(\mathbf{r}))$. We use L_1 as our objective to train the autoencoder.

The dense representation of motion vector and residual with the same spatial resolution enables us to concatenate them channel-wise as $[\mathbf{m}, \mathbf{r}]$. We can uniformly sample the time steps t to learn the joint distribution of $[\mathbf{m}, \mathbf{r}]$ with the

following objective function:

$$L = \mathbb{E}_{t, \mathbf{m}, \mathbf{r} = f(v_0), v_0 \sim \mathcal{V}, \epsilon \sim \mathcal{N}(0, 1)} [\lambda(t) \text{mse}] \quad (8)$$

$$\text{mse} = \|\epsilon - \epsilon_\theta(\mathbf{m}_t, \mathcal{E}(\mathbf{r}_t), t, \tau_\theta(v_0^0))\|^2.$$

Experiments

Datasets and Metrics

Datasets We conduct our experiment on well-known video datasets used for image-to-video generation: MHAD (Chen, Jafari, and Kehtarnavaz 2015), NATOPS (Yale Song and Davis 2011), and BAIR (Ebert et al. 2017). The MHAD human action dataset comprises 861 video recordings featuring 8 participants performing 27 different activities. This dataset encompasses a variety of human actions including sports-related actions like bowling, hand gestures such as 'draw x', daily activities like transitioning from standing to sitting, and workout exercises like lunges. For training and testing purposes, we've randomly picked 602 videos from all subjects

Method	MHAD			NATOPS		
	FVD↓	cFVD↓	sFVD↓	FVD↓	cFVD↓	sFVD↓
ImaGINator (WACV 2020)	889.48	1406.56	1175.74	721.17	1122.13	1042.69
VDM (Arxiv 2022)	295.55	531.20	398.09	169.61	410.71	350.59
LDM ₆₄ (CVPR 2022)	280.26	515.29	427.03	251.72	506.40	491.37
LFDM ₆₄ (CVPR 2023)	152.48	339.63	242.61	160.84	376.14	324.45
D-VDM₆₄	145.41	308.33	244.73	152.19	358.47	266.53
LDM ₁₂₈ (CVPR 2022)	337.43	594.34	497.50	344.81	627.84	623.13
LFDM ₁₂₈ (CVPR 2023)	214.39	426.10	328.76	195.17	423.42	369.93
ED-VDM₁₂₈ (110× speedup)	204.17	389.70	348.10	179.65	373.23	351.26

Table 1: Quantitative comparison of conditional Image-to-Video generation on MHAD and NATOPS datasets. We compare FVD, sFVD, and cFVD on 16 frames clip. The 64 and 128 in the subscript indicate that the resolution of synthesized video frames is 64×64 and 128×128 , respectively.

for the training set and 259 videos for the testing set. The NATOPS aircraft handling signal dataset encompasses 9,600 video recordings that involve 20 participants executing 24 distinct body-and-hand gestures employed for interaction with U.S. Navy pilots. This dataset features common handling signals like ‘spread wings’ and ‘stop’. We have arbitrarily chosen 6720 videos from all subjects for the training phase, while the remaining videos are for the testing phase. Detailed data preprocess methods are listed in Appendix.

Metrics For evaluation metrics of the text conditional image to video task, we follow the protocol proposed by LFDM (Ni et al. 2023) and report the Fréchet Video Distance (FVD) (Unterthiner et al. 2018), class conditional FVD (cFVD) and subject conditional FVD (sFVD) for MHAD and NATOPS datasets. FVD utilizes a pre-trained I3D (Carreira and Zisserman 2017) video classification network from the Kinetics-400 (Kay et al. 2017) dataset to derive feature representations of both real and generated videos. The cFVD and sFVD evaluate the disparity between the actual and generated video feature distributions when conditioned on the same class label y or the identical subject image x_0 , respectively. In addition, for the image-to-video task, we report the FVD score on BAIR datasets. All evaluation is conducted on 2048 randomly selected real and generated 16 frames video clips following the protocol proposed by StyleGAN-V (Karras et al. 2019).

Implementation Details

We use a conditional 3D U-Net architecture as the denoising network, and directly apply the multi-head self-attention (Cheng, Dong, and Lapata 2016) mechanism to the 3D video signal. Additionally, we use a ResNet(He et al. 2016) block to encode the first frame as a conditional feature map and provided it to ϵ_θ by the concatenation with the noise ϵ . In ED-VDM, The feature map of the first frame is downsampled $16\times$ to match the size of the motion feature, and in D-VDM the feature map remains the original size. We use the pre-trained CLIP (Radford et al. 2021) to encode text y as text embedding e , and we adopt the classifier-free guidance method in the training process. The embedding e is concatenated with the time step embedding. Detailed U-

net structures of ED-VDM and D-VDM can be found in the supplementary material.

For ED-VDM, we compress the motion vector according to the motion block size (values in the same block inside the motion vector are the same), and we employ a VAE (Romach et al. 2022) with slight KL-regularization of $1e^{-6}$ to encode the residual into a $16 \times 16 \times 16$ latent representation. Detailed model architectures are listed in the supplementary material. For different datasets, we take the middle 4 residuals out of 16 in a video clip as the training set to train our VAE model.

Baselines

We compare our approach with the most recent diffusion-based methods including MCVD (Voleti, Jolicoeur-Martineau, and Pal 2022), CCVS (Moing, Ponce, and Schmid 2021), and VDM (Ho et al. 2022b) on image-to-video tasks in BAIR datasets. We also compare our method with LFDM(Ni et al. 2023), VDM (Ho et al. 2022b), and LDM (He et al. 2022b) on MHAD and NATOPS for text condition image-to-video tasks. We collect the performance scores of the above methods from their original paper.

Main Results

Stochastic Image to Video Generation. Table 3 shows the quantitative comparison between our method and baseline methods for the image-to-video (I2V) task on the BAIR dataset with 64×64 resolution. By simply changing the target space from RGB pixels to image residuals, D-VDM improved the previous best FVD from 66.9 to 65.5. By utilizing motion vector and residual, ED-VDM achieves a high compression rate and a competitive performance. For image quality, our D-VDM method surpasses previous methods on PSNR, SSIM, and LPIPS from 16.9 to 17.6, 0.780 to 0.799, and 0.122 respectively. The image quality of the generated video by ED-VDM is slightly lower than the SOTA method but with a much higher speed advantage. We assume that due to the low image resolution, the compressed temporal latent only has a spatial size of 4×4 which can not contain enough information to restore the original video. Further experiments on high-resolution videos demonstrate the superiority of our ED-VDM method.

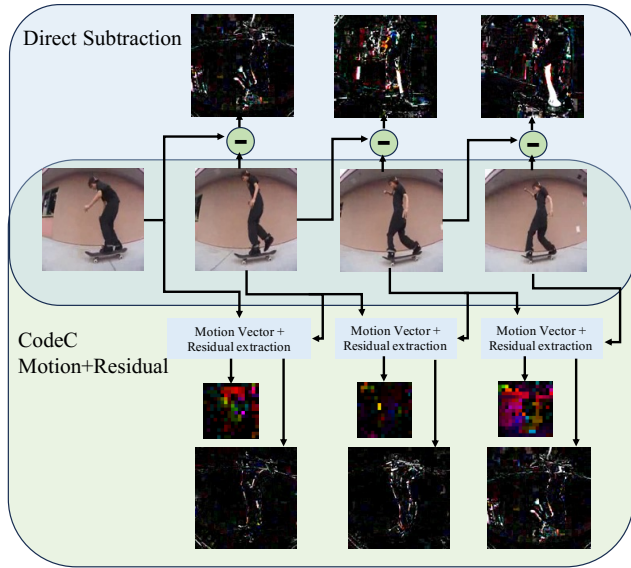


Figure 3: Different ways to represent video temporal feature between frames. Frame difference used by D-VDM, a simple technique, calculates direct frame discrepancies, encompassing both fundamental and advanced temporal alterations. D-VDM findings reveal its potency in refining the temporal consistency of a generated video. Motion Vector and Residual used by ED-VDM, as in H.264, disentangles temporal shifts into intermediate motion blocks and pixel residuals. Notably, correlated with motion vectors, these residuals offer a sparse representation with potent compression potential. In our experiments, ED-VDM attains an impressive 110x compression ratio.

Conditional Image to Video Generation. We conduct experience on NATOPS and MHAD following the protocols proposed by LFDM (Ni et al. 2023). Our D-VDM achieved remarkable results on MHAD and NATOPS at 64×64 resolutions, outperforming all previous SOTA methods. In specific, D-VDM achieves an FVD score of **145.41** on MHAD and **152.19** on NATOPS. Considering the effect of the text condition and the subject image, we further report the cFVD and sFVD scores of our method, and our method achieves SOTA performance. We conducted further experiments with the ED-VDM method on NATOPS and MHAD with larger resolutions. Our results are presented in Table 1, and ED-VDM achieved comparable results at 128 resolutions with 110 times speedup than VDM (He et al. 2022a). Specifically, our proposed ED-VDM achieved an FVD score of **204.17** on MHAD and **179.65** on NATOPS which surpasses all previous methods. For the different text conditions and subject images, our method achieves SOTA performance on both sFVD and cFVD.

For qualitative results, Figure 4 illustrates the video generation samples from our D-VDM and ED-VDM methods on BAIR, MHAD, and NATOPS datasets. The figure demonstrates that our proposed approach can generate realistic and temporally consistent videos on three datasets. With the text condition on dataset MHAD and NATOPS, generated videos

	R-FVD↓	PSNR↑	SSIM↑
128-MHAD	130.62	31.80	0.95
128-NATOPS	131.91	31.60	0.96

Table 2: The upper bound of our ED-VDM method. R-FVD score is evaluated with 2,048 samples. PSNR and SSIM are evaluated on an average of 16 frames with 100 samples.

achieve a strong correlation with the text condition. Furthermore, using ED-VDM, we can still generate high-fidelity videos with comparable quality, which leverage a 110 times training and inference efficiency.

Analysis

Reconstruction Quality

Table 2 summarizes the results of the reconstruction quality of our residual autoencoder. We use the R-FVD, which indicates FVD between reconstructions and the ground-truth real videos, peak-signal-to-noise ratio (PSNR), and structural similarity index measurement (SSIM) to evaluate the image quality with residual reconstruction. All evaluations are conducted on randomly selected reconstructed videos and real videos. Quantitative results in Figure 5 show that our residual reconstruction method achieves a small image quality degradation.

BAIR (64x64)	FVD↓	PSNR ↑	SSIM ↑	LPIPS ↓
CCVS	99.0	-	0.729	-
MCVD	89.5	16.9	0.780	-
VDM	66.9	-	-	-
D-VDM	65.5	17.6	0.799	0.122
ED-VDM (110xspeedup)	92.4	16.0	0.775	0.132

Table 3: Image-to-Video Generation Results on BAIR dataset. Our method surpasses the SOTA methods with regard to FVD score.

Method	FLOPs($\times 10^9$)	Memory (GB)
VDM (Ho et al. 2022b)	8814	11.56
LFDM (Ni et al. 2023)	627	6.69
D-VDM	8611	11.49
ED-VDM	78	3.47

Table 4: FLOPs and memory usage for our model to train on 1 batch of $16 \times 128 \times 128 \times 3$ resolution videos.

Speed Comparison

As shown in Table 4, we evaluate the FLOPs and memory consumption to train on $128 \times 128 \times 3$ video clips of different methods. Since D-VDM directly uses a 3D U-net to train on original video frames, it has approximately the same FLOPs and memory as the video diffusion model (VDM). With

BAIR
Stochastic Generation



MHAD
"cross arms in the chest"

MHAD
"baseball swing from right"

NATOPS
"All Clear"

NATOPS
"I Have Command"

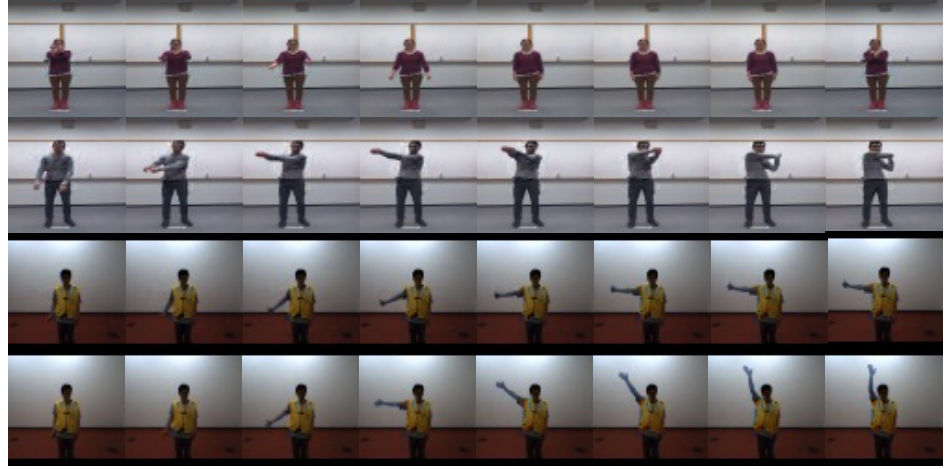
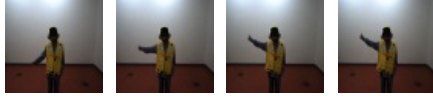


Figure 4: Selected samples on BAIR, NATOPS, and MHAD dataset. First two rows are the results of unconditionally generation results on BAIR, and the down four rows are text conditional generation results on MHAD and NATOPS. The visualization results show our method generated realistic and temporally consistent video frames.

**Reconstructed
Residual Frames**



**Ground
Truth Frames**

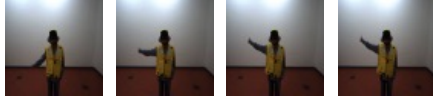


Figure 5: Video quality on residual reconstruction.

128-MHAD-101	PSNR $\uparrow$	SSIM $\uparrow$
Autoencoder	31.80	0.96
DCT	17.08	0.85

Table 5: Image quality comparison between our proposed autoencoder method and traditional DCT method.

the residual compression and motion vector, our ED-VDM model achieves a $256\times$ spatial compression rate than VDM and $\sim 110\times$, $\sim 8\times$ better computation efficiency than VDM and LFDM, respectively.

Compression Method Exploration

To compress the residual to match the dimension with the motion vector, we evaluated two methods to compress and reconstruct, including Discrete Cosine Transformation (DCT)

and autoencoder compression. Table 5 shows the image reconstruction quality of different approaches. We can see that the adopted autoencoder achieves better reconstruction quality in both metrics.

Conclusion

This paper demonstrates that transforming the target generation space from RGB pixels to temporal features can significantly improve the temporal consistency and reduce the computational efficiency of video generation. We propose Decoupled Video Diffusion Model (D-VDM), which achieves SOTA performance on various video generation tasks by decoupling the video into key frame and frame residuals. Furthermore, our proposed ED-VDM further takes advantage of the sparsity in the motion and frame compensation features to achieve comparable SOTA results with notable speedup ($110\times$). These results demonstrate the effectiveness of our decouple-based approach and open up possibilities for future work in video generation research.

Acknowledgments

This work is supported by the National Key R&D Program of China under Grant 2021YFF0602101. This work is also supported by National Natural Science Foundation(NNSF) of China under Grants 61973005 and 62373352 and in part by the InnoHk program.

References

- Bao, F.; Li, C.; Sun, J.; and Zhu, J. 2022. Why Are Conditional Generative Models Better Than Unconditional Ones? *arXiv preprint arXiv:2212.00362*.
- Bao, F.; Nie, S.; Xue, K.; Li, C.; Pu, S.; Wang, Y.; Yue, G.; Cao, Y.; Su, H.; and Zhu, J. 2023. One Transformer Fits All Distributions in Multi-Modal Diffusion at Scale.
- Blattmann, A.; Rombach, R.; Ling, H.; Dockhorn, T.; Kim, S. W.; Fidler, S.; and Kreis, K. 2023. Align your Latents: High-Resolution Video Synthesis with Latent Diffusion Models. *arXiv:2304.08818*.
- Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.
- Chen, C.; Jafari, R.; and Kehtarnavaz, N. 2015. UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *2015 IEEE International Conference on Image Processing (ICIP)*, 168–172.
- Cheng, H. K.; Tai, Y.-W.; and Tang, C.-K. 2021. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5559–5568.
- Cheng, J.; Dong, L.; and Lapata, M. 2016. Long Short-Term Memory-Networks for Machine Reading. In *2016 Conference on Empirical Methods in Natural Language Processing*, 551–561. Association for Computational Linguistics.
- Choi, J.; Kim, S.; Jeong, Y.; Gwon, Y.; and Yoon, S. 2021. ILVR: Conditioning Method for Denoising Diffusion Probabilistic Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14367–14376.
- Ebert, F.; Finn, C.; Lee, A. X.; and Levine, S. 2017. Self-Supervised Visual Planning with Temporal Skip Connections. *CoRR*, abs/1710.05268.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative Adversarial Networks. *Commun. ACM*, 63(11): 139–144.
- Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; and Guo, B. 2022. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10696–10706.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.
- He, Y.; Yang, T.; Zhang, Y.; Shan, Y.; and Chen, Q. 2022a. Latent Video Diffusion Models for High-Fidelity Video Generation with Arbitrary Lengths. *arXiv preprint arXiv:2211.13221*.
- He, Y.; Yang, T.; Zhang, Y.; Shan, Y.; and Chen, Q. 2022b. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*.
- Ho, J.; Chan, W.; Saharia, C.; Whang, J.; Gao, R.; Gritsenko, A. A.; Kingma, D. P.; Poole, B.; Norouzi, M.; Fleet, D. J.; and Salimans, T. 2022a. Imagen Video: High Definition Video Generation with Diffusion Models. *ArXiv*, abs/2210.02303.
- Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33: 6840–6851.
- Ho, J.; Salimans, T.; Gritsenko, A.; Chan, W.; Norouzi, M.; and Fleet, D. J. 2022b. Video diffusion models. *arXiv:2204.03458*.
- Huang, L.; Liu, Y.; Wang, B.; Pan, P.; Xu, Y.; and Jin, R. 2021. Self-supervised video representation learning by context and motion decoupling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13886–13895.
- HyvärinenAapo. 2005. Estimation of Non-Normalized Statistical Models by Score Matching. *Journal of Machine Learning Research*.
- Karras, T.; Aittala, M.; Aila, T.; and Laine, S. 2022. Elucidating the Design Space of Diffusion-Based Generative Models. In *Proc. NeurIPS*.
- Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2019. Analyzing and Improving the Image Quality of StyleGAN. *CoRR*, abs/1912.04958.
- Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; et al. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.
- Le Gall, D. 1991. MPEG: A Video Compression Standard for Multimedia Applications. *Commun. ACM*, 34(4): 46–58.
- Mei, K.; and Patel, V. M. 2022. VIDM: Video Implicit Diffusion Models.
- Meng, C.; Song, Y.; Song, J.; Wu, J.; Zhu, J.-Y.; and Ermon, S. 2021. SDEdit: Image Synthesis and Editing with Stochastic Differential Equations. *arXiv: Computer Vision and Pattern Recognition*.
- Moing, G. L.; Ponce, J.; and Schmid, C. 2021. CCVS: Context-aware Controllable Video Synthesis. In *NeurIPS*.
- Ni, H.; Shi, C.; Li, K.; Huang, S. X.; and Min, M. R. 2023. Conditional Image-to-Video Generation with Latent Flow Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18444–18455.
- Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2023. GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models.
- Poole, B.; Jain, A.; Barron, J. T.; Mildenhall, B.; Research, G.; and Berkeley, U. C. 2023. DREAMFUSION: TEXT-TO-3D USING 2D DIFFUSION.
- 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. *arXiv:2103.00020*.

- Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2023. Hierarchical Text-Conditional Image Generation with CLIP Latents.
- Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-Shot Text-to-Image Generation. *arXiv:2102.12092*.
- Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10684–10695.
- Singer, U.; Polyak, A.; Hayes, T.; Yin, X.; An, J.; Zhang, S.; Hu, Q.; Yang, H.; Ashual, O.; Gafni, O.; Parikh, D.; Gupta, S.; and Taigman, Y. 2022. Make-A-Video: Text-to-Video Generation without Text-Video Data.
- Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2256–2265. PMLR.
- Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.
- Springenberg, J. T. 2015. Unsupervised and Semi-supervised Learning with Categorical Generative Adversarial Networks. *arXiv: Machine Learning*.
- Unterthiner, T.; van Steenkiste, S.; Kurach, K.; Marinier, R.; Michalski, M.; and Gelly, S. 2018. Towards Accurate Generative Models of Video: A New Metric & Challenges. *ArXiv*, abs/1812.01717.
- Villegas, R.; Babaeizadeh, M.; Kindermans, P.-J.; Moraldo, H.; Zhang, H.; Saffar, M. T.; Castro, S.; Kunze, J.; and Erhan, D. 2022. Phenaki: Variable Length Video Generation From Open Domain Textual Description.
- Voleti, V.; Jolicoeur-Martineau, A.; and Pal, C. 2022. MCVD: Masked Conditional Video Diffusion for Prediction, Generation, and Interpolation. In *(NeurIPS) Advances in Neural Information Processing Systems*.
- Wu, C.-Y.; Zaheer, M.; Hu, H.; Manmatha, R.; Smola, A. J.; and Krähenbühl, P. 2018. Compressed video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6026–6035.
- Yale Song, D. D.; and Davis, R. 2011. Tracking Body and Hands For Gesture Recognition: NATOPS Aircraft Handling Signals Database. In *In Proceedings of the 9th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2011)*.
- Yang, Z.; and Yang, Y. 2022. Decoupling features in hierarchical propagation for video object segmentation. *Advances in Neural Information Processing Systems*, 35: 36324–36336.
- Yu, S.; Sohn, K.; Kim, S.; and Shin, J. 2023. Video Probabilistic Diffusion Models in Projected Latent Space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Yu, S.; Tack, J.; Mo, S.; Kim, H.; Kim, J.; Ha, J.-W.; and Shin, J. 2022. Generating Videos with Dynamics-aware Implicit Generative Adversarial Networks. In *International Conference on Learning Representations*.