Denoising Reuse: Exploiting Inter-frame Motion Consistency for Efficient Video Generation

Chenyu Wang*, Shuo Yan*, Yixuan Chen*, Xianwei Wang, Yujiang Wang, Mingzhi Dong, Xiaochen Yang, Dongsheng Li, *Member, IEEE,* Rui Zhu, David A. Clifton, Robert P. Dick, *Senior Member, IEEE,* Qin Lv, Fan Yang, Tun Lu, *Member, IEEE,* Ning Gu, *Member, IEEE,* and Li Shang†, *Member, IEEE,*

***Abstract*—Denoising-based diffusion models have attained impressive image synthesis; however, their applications on videos can lead to unaffordable computational costs due to the per-frame denoising operations. In pursuit of efficient video generation, we present a Diffusion Reuse MOtion (Dr. Mo) network to accelerate the video-based denoising process. Our crucial observation is that the latent representations in early denoising steps between adjacent video frames exhibit high consistencies with motion clues. Inspired by the discovery, we propose to accelerate the video denoising process by incorporating lightweight, learnable motion features. Specifically, Dr. Mo will only compute all denoising steps for base frames. For a non-based frame, Dr. Mo will propagate the pre-computed based latents of a particular step with inter-frame motions to obtain a fast estimation of its coarse-grained latent representation, from which the denoising will continue to obtain more sensitive and fine-grained representations. On top of this, Dr. Mo employs a meta-network named Denoising Step Selector (DSS) to dynamically determine the step to perform motion-based propagations for each frame, ensuring the correct transformation of multi-granularity visual features. Extensive evaluations on video generation and editing tasks indicate that Dr. Mo delivers widely applicable acceleration for diffusion-based video generations while effectively retaining the visual quality and style. Video generation and visualization results can be found at https://drmo-denoising-reuse.github.io.**

***Index Terms*—Video Generation, Diffusion Models, Computational Efficiency**

I. Introduction

DIFFUSION models such as Denoising Diffusion Probabilistic Models (DDPMs) [1] and Video Diffusion Models (VDMs) [2] have demonstrated impressive capabilities to generate high-fidelity videos. However, the superior visual qualities come at the cost of computation burdens primarily associated with the denoising operation of numerous steps [3, 4]. This is cost-prohibitive for videos; applying diffusion models on a per-frame basis imposes computational demands that increase linearly with frames, undermining the generation of long-duration videos [2] and hindering the deployment of video generation models in practical applications.

This work presents a novel framework to dramatically accelerate diffusion-based video generation via employing motion dynamics in the latent space of the denoising process. We embark on a comprehensive investigation of the video generation process to illustrate our insights. As shown in Figure 1 (left), the diffusion model applies incremental noise reduction to gradually recover visual features from gaussian white noise. Due to the nature of visual features, which exhibit higher low-frequency energy and lower high-frequency energy in the frequency domain, the model follows a coarse-to-fine pattern during the denoising process. Compared to the later denoising steps, the visual features obtained in the earlier steps are coarser, more semantic, and exhibit stronger inter-frame consistency. This opens up the possibility of using a faster estimation method, such as motion cues in the video, to acquire the latent representations.

We further inspect the trend of inter-frame motion dynamics, computed by the normalized mutual information (NMI) between learned motion matrices detailed in Section III-B, across the denoising steps, as portrayed in Figure 1 (right). Intuitively, higher NMI values typically indicate more consistent inter-frame motion dynamics in the denoising step. We can see from the curve that the motion consistencies are generally high across numerous denoising steps, especially those operating on coarse-grained features, which unveil the consistent nature of motion information across denoising steps. Those observations inspire an approach to accelerate the video-based denoising process. The early-stage denoising latent representations of a video frame can be reused with easy-to-compute motion features to efficiently obtain the latent representations in subsequent frames, since those coarse-grained representations can be more tolerant to the inaccurateness from fast estimations, and the high-frequency noise generated by feature distortion

* Equal contribution.
† Corresponding author.
C. Wang, S. Yan, Y Chen, X Wang, F. Yang, T. Lu, N. Gu, L. Shang are with School of Computer Science, Fudan University, Shanghai, China, Shanghai Key Laboratory of Data Science, Fudan University, Shanghai, China, and Shanghai Institute of Intelligent Electronics & Systems, Shanghai, China.
M. Dong is with the University of Bath, Bath, UK, and also with the School of Computer Science, Fudan University, Shanghai, China.
Y. Wang is with Oxford Suzhou Centre for Advanced Research, Suzhou, China.
X. Yang is with School of Mathematics and Statistics, University of Glasgow, UK.
R. Zhu is with Bayes Business School, City, University of London, UK.
D. A. Clifton is with Oxford Suzhou Centre for Advanced Research, Suzhou, China, and also with Department of Engineering Science, University of Oxford, Oxford, UK.
Q. Lv is with University of Colorado Boulder, Boulder, CO, USA.
R. P. Dick is with Department of Electrical Engineering and Computer Science College of Engineering, University of Michigan, Ann Arbor, MI, USA.
D. Li is a senior researcher with Microsoft Research Asia, Shanghai, China and an adjunct professor with School of Computer Science, Fudan University, Shanghai, China.

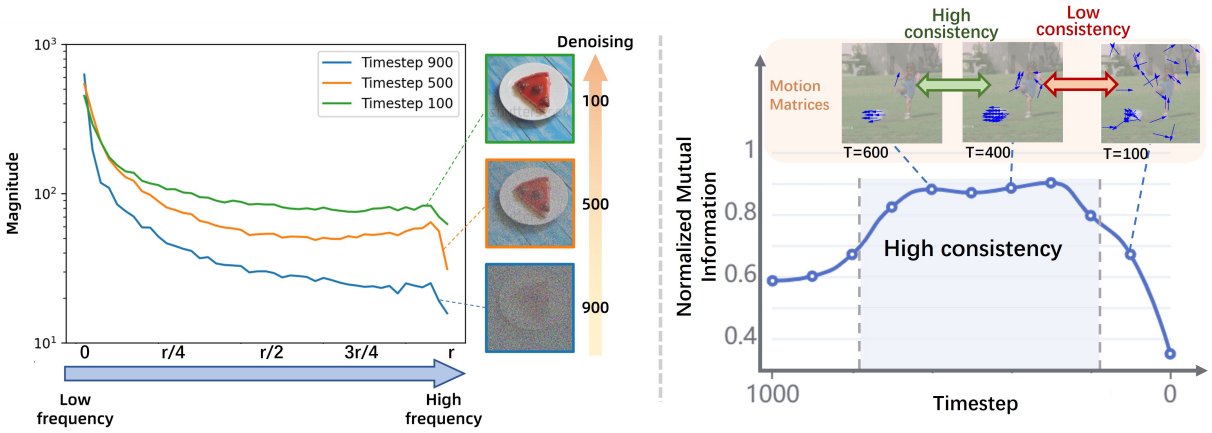


Fig. 1. Left: The spectrum illustrates an increase in high-frequency signals during the denoising process, from steps 900 to 100. Right: Statistics of NMI curves between neighboring frames of 1000 different videos sampled in webvid, which show that NMI scores are generally higher in the middle of the denoising process, indicating that the motion dynamics maintains a high degree of consistency in the middle of the denoising process.

can also be eliminated in the subsequent denoising process.

Latent representations in late denoising steps harbour more fine-grained feature crucial to the generated images' quality. Therefore, preserving the precision of those representations can be essential to attaining visual quality, and thus, denoising operations are indispensable. Although the denoising process in diffusion models is stochastic, leading to no clear boundary between coarse-grained and fine-grained features, they can still be distinguished based on their probability distributions. To handle these two types of features separately, it is necessary to identify an appropriate transition step. Those denoising steps before it could be skipped to save computations as its latent representation can be directly and rapidly estimated by reusing the pre-computed one with motion features. After the transition step, representations will still go through denoising operations to retain the desirable visual quality. The choice of the transition step depends on the magnitude of motion features in the video. A transition step that is too early or too late will affect the visual quality of the generated result.

In pursuit of efficient video generation, we devise a new Diffusion Reuse MOtion (Dr. Mo) network that significantly accelerates the video-base denoising by incorporating inter-frame motion clues. Dr. Mo consists of three major modules, a denoising-based diffusion backbone network such as the stable diffusion [5], a motion network to yield inter-frame motion dynamics, and a meta-network dubbed the Denoising Step Selector (DSS) to determine a proper transition step. Dr. Mo starts by applying the backbone diffusion network to a base video frame to obtain its latent representations at each denoising step, referred to as the base latents. For a non-base frame, DSS will be involved in selecting a proper transition step, and the motion network will be utilised to obtain motion features. The latent representation on the transition step will be obtained by warping the pre-computed base noises of the same step with motion features, which could save the computations of those earlier denoising steps. The estimated latent representation will be fed into the backbone diffusion network to complete the resting denoising steps and obtain the final frame prediction. Moreover, we have developed several objective functions to train Dr. Mo, including the motion and DSS networks.

Our evaluations on the UCF-101 [6] and MSR-VTT [7] datasets have demonstrated Dr. Mo's superior video quality and semantic alignment over state-of-the-art baselines. Notably, Dr. Mo effectively accelerates the generation of 16-frame 256×256 videos 5.4 times faster compared with Latent-Shift [8] while maintaining 96% of the Inception Score (IS) [9] and achieving improved Fréchet Video Distance (FVD) [10]. Compared with SimDA [11] on generating 16-frame 512×512 videos, Dr. Mo works 2.2 times faster. As an easy-to-implement and flexible module, Dr. Mo can also be effortlessly integrated into the framework of various diffusion-based video generation approaches, providing a more than 2x acceleration in video generation with comparable visual quality and stylistic coherence.

In summary, our work makes the following contributions:

1) We intuitively reveal that inter-frame motion dynamics are highly consistent across the denoising steps, a critical insight for accelerating diffusion with inter-frame motions.
2) We present Dr. Mo, an effective method to accelerate the video-based denoising process, which could speed up video generation by around 2x with matching or improved visual qualities. Dr. Mo also generalises well to various diffusion models.
3) We develop approaches to learning a lightweight motion network to generate proper motion features and a meta-network to dynamically determine the transition steps for correctly processing multi-granularity features.

II. Related Work

A. Diffusion-based Image Generation

Diffusion models have achieved state-of-the-art results in text-to-image generation [12, 13, 14], garnering significant attention from both the academic community and industry. GLIDE [15] introduced text conditioning, demonstrating that

classifier guidance can yield more satisfying results. DALL-E2 [16] improved text-image alignment by leveraging the CLIP [17] joint feature space, which allows users to provide a text prompt and generate images of unprecedented quality. Imagen [12] combined large language models with a cascade architecture to produce realistic outcomes. The latent diffusion model [5], also known as Stable Diffusion (SD), shifts the diffusion process into the latent space of an autoencoder, significantly enhancing efficiency and becoming the most widely used diffusion backbone model to date. Most of the video generation models we present are developed based on SD.

The latent space of diffusion models has also been extensively studied. For instance, EmerDiff [18] utilized the latent space of pre-trained diffusion models on generative tasks to achieve zero-shot segmentation. BE-Cycle [19] enabled fine-grained semantic editing of images by manipulating the diffusion model's latent space. In the SCFM [20] framework, diffusion models are equipped with latent and noise-guided modules to control precise positioning and pose in human photo generation. Que et al [21]. demonstrated sketch-to-photo synthesis using pre-trained diffusion models. Moreover, Dual-Cycle Diffusion [22] and RDVR+ [23] leveraged diffusion models for semantic-level video compression, enabling the reconstruction of original footage and detecting anomalies within the latent space. These works collectively suggest that diffusion models possess a semantic-level latent space, which allows for efficient motion modeling and precise frame control during video generation.

B. Diffusion-based Video Generation

Recent advances in diffusion-based models [2, 24, 25, 26] have integrated spatiotemporal operations into traditional image-based frameworks, producing high-quality videos and breaking the dominance of GANs in the field of video generation [27, 28, 29]. However, their reliance on iterative denoising processes makes them computationally expensive and unnecessarily slow. To simplify video generation, recent research has turned to latent space-based models [11, 30, 31, 32], particularly latent diffusion models [1, 4]. For instance,SimDA [11] maintains the parameter of text-to-image diffusion model fixed and utilizes lightweight spatial adapter and temporal adapter for learning visual information and temporal relationships in videos. For motion information learning, AnimateDiff [31] adds a novel motion module to base diffusion model to learn motion dynamics. LVDM [33] and LaVie [34] generate sparse video patterns and interpolate intermediate latents, but do not explicitly model motion information. Latent-Shift [8] uses feature maps from adjacent frames to facilitate motion learning without extra parameters, while Text2Video-Zero [35] employs predefined direction vectors to introduce motion dynamics, yet struggles with temporal consistency. VideoLCM [36] employs a teacher-student framework to distill consistency to minimize steps. However, it requires fine-tuning the complete diffusion process for each frame, taking 10s to generate $16\times256\times256$ frames. In contrast, our approach takes only 4.35s with 50 steps using DDIM [4]. VidRD [37] also reuses latent features from previously generated clips does not adapt the number of reuse steps across frames, limiting its efficiency.

To the best of our knowledge, we are the first to apply the sparse representation brought by inter-frame consistency in video generation, and improve generation efficiency through efficient feature reuse. Furthermore, we have conducted an in-depth analysis of the representational characteristics within diffusion models, explaining the theory behind motion representation extraction and the use of compressed representations in diffusion, offering new insights and perspectives.

C. Motion Estimation

The core of motion estimation is to find the motion of pixels or feature points between images or video frames. Existing methods typically include Block Matching Algorithm (BMA) [38], Optical Flow [39, 40, 41, 42, 43, 44], and Feature Tracking [45, 46, 47, 48, 49, 50].

BMA is a traditional, non-deep learning method. The core idea is to divide an image into multiple small blocks and find the corresponding position in the next frame for each block [38]. It is computationally simple and efficient, and is widely used in video compression, object tracking, and motion detection, among other fields. Although BMA is a relatively simple traditional algorithm, its core idea has been widely applied in video coding and compression fields, inspiring higher-performance works in various scenarios [51, 52, 53].

Optical flow estimates pixel motion across consecutive frames based on local intensity changes. Traditional methods rely on handcrafted features and are computationally expensive. Recently, deep learning-based methods, offering better accuracy and efficiency, have become dominant. FlowNet [39] introduced two key architectures: FlowNet-S, which stacks images on the channel level, and FlowNet-C, which computes correlations at the feature level. FlowNet 2.0 [40] stacked these modules and added optical flow residual prediction, surpassing traditional methods. PWC-Net [41] combined pyramid processing, warping, and cost volume techniques for improved performance. RAFT [42] introduced cyclic full-pair field transformations and update operators, further enhancing accuracy. Later works have focused on unsupervised methods [43, 54], as well as methods using masks [44] or depth [55] estimation.

Feature tracking estimates object motion by detecting and tracking feature points across frames. Traditional image matching involves feature detection, description, matching, and geometric transformation [56]. Deep learning methods focus on improving these stages and are divided into Detector-based and Detector-free approaches. Detector-based methods detect and describe key points, with traditional techniques like SIFT [45] and ORB [57] relying on handcrafted designs. Data-driven methods, such as LIFT [58], have emerged, with four main strategies: Detect-then-Describe [45, 58], Joint Detection and Description [46, 59], Describe-then-Detect [60, 61], and Graph-Based [48, 62], improving robustness to viewpoint and lighting changes. Detector-free methods eliminate key point detection and description, leveraging rich contextual information to identify repeatable points. Early CNN-based methods, like NCNet [47], evolved into Transformer-based [63, 64] and patch-based methods [49, 65].

In recent years, diffusion models have gradually become mainstream as generative visual models. Diffusion models naturally possess multi-scale and multi-resolution visual features, and the semantic information they contain supports semantic segmentation and feature matching. These properties make diffusion models potentially useful for motion estimation, although the exploration of motion estimation based on diffusion is still underdeveloped in existing research.

III. Motion Dynamics in Diffusion Model

This section analyzes motion dynamics throughout the coarse- to fine-grained visual feature generation process. We find that motion dynamics are consistent in the majority of denoising steps and the optimal number of reuse steps is frame dependent. These phenomena motivate us to adaptively reuse denoising steps across frames for efficient video generation.

A. Motion Dynamics

In this study, we employ the Stable Diffusion (SD) model as the backbone model for generating videos if not specifically emphasized. Consider a video comprising F frames, denoted by $I = [I^1, \ldots, I^F]$. Initially, each frame I^i is encoded into a latent space representation $\mathbf{z}^i$. We employ the DDPM approach with $T = 1000$ denoising steps to recover the original frames. The denoising process recovers each frame from step T to 1. $\mathbf{z}_t^i$ represents the latent state of frame i at timestep t, where t indicates the denoising timestep and i indicates the frame number within the video sequence.

To analyze the inter-frame motion dynamics for generating coherent videos using a diffusion model, we introduce the concept of latent residual to represent the change in latent features between two steps, denoted as:

$$\delta \mathbf{z}_t^i := \mathbf{z}_{t-1}^i - \mathbf{z}_t^i. \tag{1}$$

This difference can be regarded as the feature revealed (or noise removed) due to the denoising process. Consequently, the latent representation at denoising step t for frame i can be reconstructed by summing the following residuals: $\mathbf{z}_t^i = \mathbf{z}_T^i + \sum_{k=t+1}^{T} \delta \mathbf{z}_k^i$, where $\mathbf{z}_T^i$ denotes the initial noisy image at the start of the reverse denoising process.

Next, we introduce the concept of a transformation operation between frames (denoted as g) to characterize inter-frame motion dynamics in latent residuals corresponding to the same denoising step. Considering frames i and j, g_ϕ^t transforms $\delta \mathbf{z}_t^i$ to match $\delta \mathbf{z}_t^j$ governed by minimizing the transformation error, as expressed by

$$\min_{\phi} \|\delta \mathbf{z}_t^j - g_\phi^t(\delta \mathbf{z}_t^i)\|_1, \quad \text{where } i < j. \tag{2}$$

Drawing inspiration from optical flow techniques [66], we propose to represent motion dynamics between frames using function $\mathcal{C}(\cdot, \cdot)$ to compute motion matrix $\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}$. This motion matrix describes the temporal relations between the residual $\delta \mathbf{z}_t^i$ and $\delta \mathbf{z}_t^j$ at the same denoising steps t, defined as:

$$\begin{aligned} g_\phi^t(\delta \mathbf{z}_t^i) &= \mathbf{M}_{\delta \mathbf{z}_t}^{i,j} \times \delta \mathbf{z}_t^i, \\ \text{where} \quad \mathbf{M}_{\delta \mathbf{z}_t}^{i,j} &= \mathcal{C}(\delta \mathbf{z}_t^i, \delta \mathbf{z}_t^j) = \frac{\delta \mathbf{z}_t^i \times (\delta \mathbf{z}_t^j)^\top}{\|\delta \mathbf{z}_t^i\| \|\delta \mathbf{z}_t^j\|}. \end{aligned} \tag{3}$$

Here, $\mathcal{C}(\cdot, \cdot)$ denotes a motion modeling function based on the cosine-similarity computation [67, 68], $\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}$ can be regarded as a heatmap, indicating the moving transition relations between latent features. Details are provided in Section IV-B.

B. Temporal Consistency of Latent Motion Dynamics

This subsection defines and quantifies the temporal consistency of latent motion dynamics.

Definition 1 (Step-wise Temporal Consistency of Motion Dynamics) Given motion matrices $\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}$ and $\mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j}$ between frames i and j at denoising timestep t and $t+1$, the temporal consistency of motion dynamics is defined as the degree of similarity between the two matrices.

To quantify this consistency, we use Normalized Mutual Information (NMI), defined as:

$$\text{NMI}(\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}, \mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j}) = \frac{I(\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}; \mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j})}{\sqrt{H(\mathbf{M}_{\delta \mathbf{z}_t}^{i,j})} \sqrt{H(\mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j})}}, \tag{4}$$

where $\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}$ and $\mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j}$ are motion matrices between frames i and j at denoising timestep t and $t+1$, respectively. I represents mutual information and H denotes entropy. By measuring the mutual information between motion matrices at different timesteps, NMI quantifies the predictive information about $\mathbf{M}_{\delta \mathbf{z}_{t+1}}^{i,j}$ from $\mathbf{M}_{\delta \mathbf{z}_t}^{i,j}$. High NMI values indicate a strong consistency of motion dynamics. To provide a more intuitive understanding, we have paired NMI values with visualizations of the motion matrices in Figure 3.

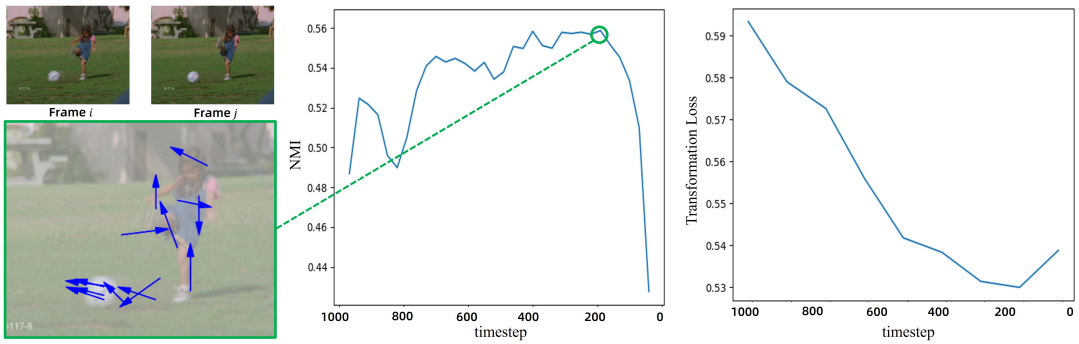
As illustrated in Figure 2, motion consistency exists throughout most steps of the diffusion process. Specifically, from the timestep 800 to 200, i.e., 60% of the denoising process, the data exhibits high NMI values and a decline in transformation errors, indicating consistent and reliable motion predictions. This consistency primarily stems from the presence of coarse-grained, semantically rich latent features that enhance the modeling of motion dynamics. In contrast, in the late denoising steps, from 200 to 1, the resulting boundaries and fine-grained features become relatively complex and semantically less meaningful. This leads to decreased predictability and a lower NMI score. These findings demonstrate the potential for reusing denoising steps across frames and reducing feature generation redundancy, which significantly enhances computational efficiency and accelerates video generation. Moreover, it allows simple control over the tradeoffs between efficiency and quality.

IV. Dr. Mo: Denoising Reuse for Efficient Video Generation

This section presents Dr. Mo, a diffusion reuse motion network that captures and uses inter-frame motion features to accelerate video latent generation in diffusion models.

A. Overview

The overall model flow of Dr. Mo is illustrated in Figure 4. Dr. Mo consists of two main components: the Motion Transformation Network (MTN) and Denoising Step Selector



(a) Visualization of Motion Matrix $\mathcal{M}^{i,j}_{z_{200}}$ (b) Normalized Mutual Information (NMI) between Frame i and Frame j (a) Transformation Loss $\mathcal{L}^{visual}_{z_t}$ between Frame i and Frame j

Fig. 2. Motion visualization at step 200 accurately captures the movement trends of patch features. At this step, the motion dynamics show consistency with low transformation errors, indicating the potential for reusing steps between step 1000 and 200.

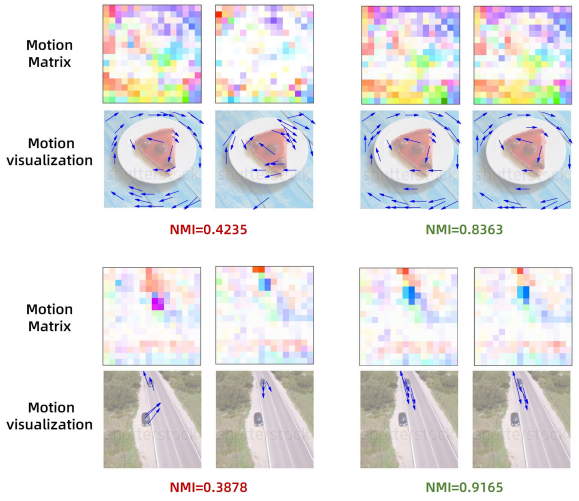


Fig. 3. Visualization of motion matrix with different Normalized Mutual Information (NMI).

(DSS). The MTN first extracts motion matrices at different timesteps from two base frames and provides the motion sequence to DSS. As shown in the right part of Figure reffig:model, DSS decides the transition step at which to switch from motion-based propagation to denoising, ensuring that features at different granularities are properly handled. Once the transition step is determined, as shown in the left part of Figure 4, MTN will predict future motion matrices based on the extracted motion matrices at that timestep, and perform feature transformation between frames to quickly estimate the coarse-grained visual features. Subsequently, as the denoising step progresses from t^* to 1, each frame goes through the original denoising process of the diffusion backbone, aiming to supplement fine-grained visual features and optimize visual quality, ultimately generating the video.

B. Motion Transformation Network

Motion Matrix Construction. In diffusion models, the denoising process of the latent space is typically performed by a U-Net module. The outputs of U-Net represent the predicted noise to be removed from $\mathbf{z}_t$ to recover $\mathbf{z}_{t-1}$. Thus, the intermediate feature of U-Net provides estimates of the residuals between these steps. Furthermore, recent studies have demonstrated that intermediate diffusion features extracted from U-Net can capture coarse- and fine-grained semantic information [18, 69, 70, 71]. Therefore, we use the latent representations from the U-Net to construct the motion matrix.

Given two video frames i and j, we extract features from multiple blocks $[b_1, \ldots, b_k]$ of the U-Net at denoising timestep t. Here, $b.$ represents the block index within the U-Net architecture. The features, denoted as $\delta\mathbf{z}^i_t[b_k]$ and $\delta\mathbf{z}^j_t[b_k]$.

Considering that the motion matrices extracted from different modules are at different spatial levels, we process these motion matrices accordingly. For layers with larger downsampling factors, the features contain richer information, thus reducing the likelihood of erroneous matching. These motion trends are typically highly confident, but due to the downsampling, spatial precision is lost, and the motion features may lack fine detail. For these layers, we merge them with higher-resolution visual features by adding position embedding and use multi-layer CNNs and self-attention for encoding, followed by cross-attention to fuse and upsample the features. This helps refine the boundaries of motion representations by leveraging visual features. For layers with smaller downsampling factors in the U-Net, which contain more detailed motion features but often fail to cover the entire object and are more focused on the edges, we stack them with visual features of the same resolution. These features are then processed through CNN and self-attention to propagate the motion information, ultimately leading to object-level motion representations.

These motion matrices are then aggregated by a multi-layer perceptron (MLP) to construct a multi-scale motion matrix:

$$\begin{aligned} \mathbf{M}^{i,j}_{\delta\mathbf{z}_t} &= g_{\phi_2}([\mathbf{M}^{i,j}_{\delta\mathbf{z}_t}[b_1], \ldots, \mathbf{M}^{i,j}_{\delta\mathbf{z}_t}[b_k]]), \\ \text{where } \mathbf{M}^{i,j}_{\delta\mathbf{z}_t}[b_k] &= \mathcal{C}(g_{\phi_1}(\delta\mathbf{z}^i_t[b_k]), g_{\phi_1}(\delta\mathbf{z}^j_t[b_k])). \end{aligned} \tag{5}$$

where ϕ_1 and ϕ_2 denote the parameters of the convolutional network and the MLP. Although the MLP is a relatively simple network, it effectively combines both linear weighting and nonlinear selection, making it well-suited to capture nonlinear

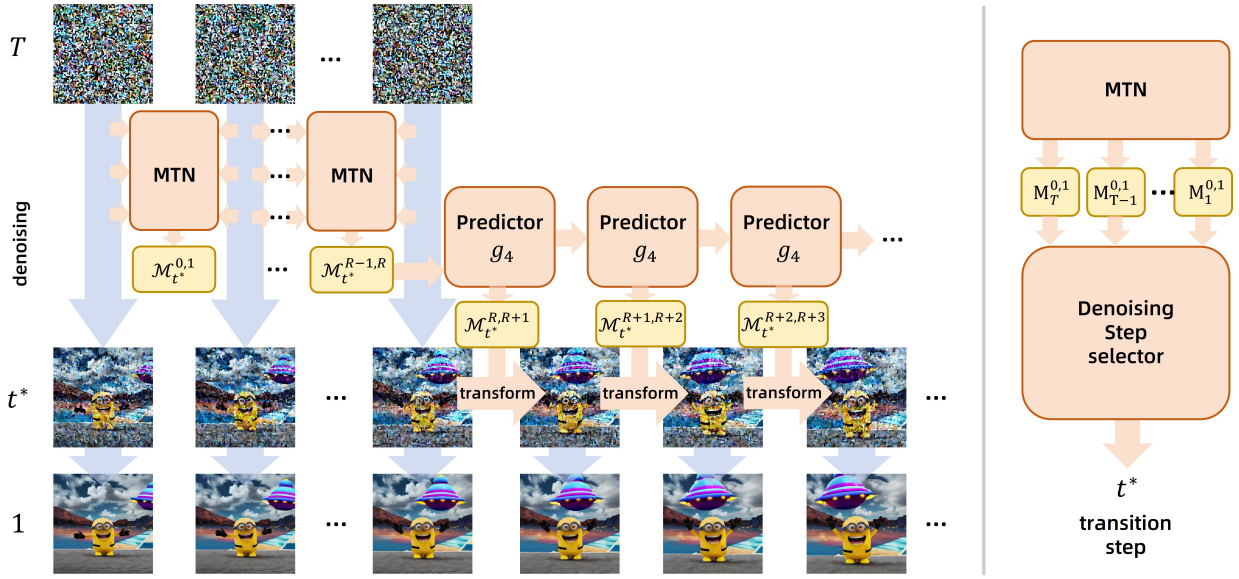


Fig. 4. Dr. Mo consists of two main components: the Motion Transformation Network (MTN) and Denoising Step Selector (DSS). MTN learns motion matrices from semantic latents extracted from U-Net and predicts motion matrices for future frames to generate coarse-grained latent representations. The DSS is a meta-network that determines the appropriate transition step (denoted as t^*) for switching from motion-based propagations to denoising, based on the motion characteristics of the video. After the transition step, those coarse-grained latent representations is processed by the rest of the diffusion model for video generation.

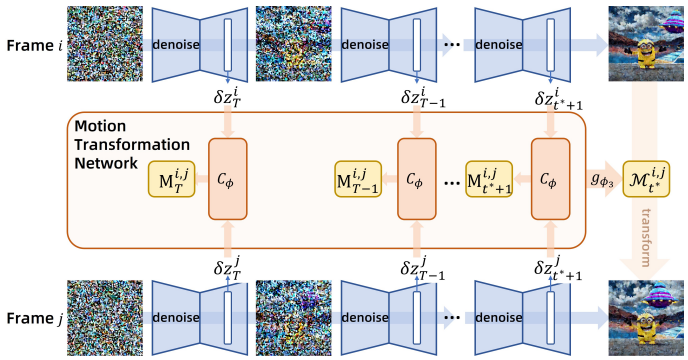


Fig. 5. The Motion Transformation Network (MTN) extracts motion matrices at different scales from the latent variables inside the U-Net during the diffusion denoising process, and aggregates them to obtain motion features that can transform the latent variables across frames.

relationships and handle diverse scenarios in motion feature aggregation.

Motion Learning Objectives. The first learning objective is to minimize the transformation loss between latent variables $\delta\mathbf{z}_t^i$ and $\delta\mathbf{z}_t^j$ at each denoising step:

$$\mathcal{L}_{\delta\mathbf{z}}^{\text{visual}} = \sum_{i,j,t} ||\mathbf{M}_{\delta\mathbf{z}_t}^{i,j} \times \delta\mathbf{z}_t^i - \delta\mathbf{z}_t^j||_1. \tag{6}$$

This computation of motion matrices with respect to the residual latents aids in modeling motion consistency. The motion sequence $\{\mathbf{M}_{\delta\mathbf{z}_t}^{i,j}\}_{t=1}^{T}$ is an input to DSS that facilitates the analysis of optimal transformation timesteps for frame i and j. Additionally, this sequence aids in approximating the surrogate matrix which we will use to transform inter-frame latents.

Given the intermediate denoising step ($t^* \in [T]$) switching from motion-based propagations to denoising (further details are provided in the subsequent section), the next task of MTN is to approximate the surrogate matrix $\mathcal{M}_{\mathbf{z}_t^*}^{i,j}$, by aggregating the motion dynamics captured within the denoising process from step T to t^*. Given the consistency observed in motion dynamics throughout most diffusion steps, $\mathcal{M}_{\mathbf{z}_t^*}^{i,j}$ can be approximated by aggregating motion dynamics from step T to step t^*. Using an MLP, g_{ϕ_3}, this process is mathematically represented as:

$$\mathcal{M}_{\mathbf{z}_t^*}^{i,j} = g_{\phi_3}(\mathbf{M}_{\delta\mathbf{z}_{t^*}}^{i,j}, \mathbf{M}_{\delta\mathbf{z}_{t^*+1}}^{i,j}, \ldots, \mathbf{M}_{\delta\mathbf{z}_T}^{i,j}). \tag{7}$$

The second learning objective is to ensure accurate inter-frame transformations using the surrogate matrix, formulated as:

$$\begin{aligned}\mathcal{L}_{\mathbf{z}}^{\text{visual}} &= \sum_{i,j} || \sum_{k=t^*}^{T} (\mathbf{M}_{\delta\mathbf{z}_k}^{i,j} \times \delta\mathbf{z}_k^i) - \sum_{k=t^*}^{T} \delta\mathbf{z}_k^j||_1 \\ &\approx \sum_{i,j} ||\mathcal{M}_{\mathbf{z}_t^*}^{i,j} \times \sum_{k=t^*}^{T} \delta\mathbf{z}_k^i - \sum_{k=t^*}^{T} \delta\mathbf{z}_k^j||_1 \\ &= \sum_{i,j} ||\mathcal{M}_{\mathbf{z}_{t^*}}^{i,j} \times \mathbf{z}_{t^*}^i - \mathbf{z}_{t^*}^j||_1.\end{aligned} \tag{8}$$

Equation 8 defines a learning objective function, aimed at ensuring the accuracy of the surrogate matrix $\mathcal{M}$ when used to transform inter-frame latent features.

To achieve the transformation from frame i to frame j, for each denoising timestep t from T to t^* of frame i's latent residual ($\delta\mathbf{z}_t^i$), we apply a motion matrix $\mathbf{M}_{\delta\mathbf{z}_k}^{i,j}$. $\mathbf{M}_{\delta\mathbf{z}_k}^{i,j}$ learns the temporal relations between latent residual of frame i and j. To closely align the transformation result with the ground truth latent residual of frame j, we utilize the L_1 norm loss function. This learning objective function represents an ideal scenario for the correct transformation of inter-frame latent features.

The first approximately equal sign represents that approximation of the motion matrices sequence $\{\mathbf{M}_t\}_{t=t^*}^{T}$ and the surrogate matrix $\mathcal{M}$. We use g_{ϕ_3} to aggregate the motion matrices sequence $\{\mathbf{M}_t\}_{t=t^*}^{T}$ and we approximate these to derive the surrogate matrix $\mathcal{M}$. Thus $\mathcal{M}$ captures the motion transformation of the cumulative latent residuals throughout the denoising process from T to t^*. We justify this approximation based on the observed consistency of motion dynamics throughout most of the diffusion process, as detailed in Section III.

The second equal symbol comes from the concept of latent residual introduced in our Equation 1. During the denoising process, the sum of the latent residual from T to t^* can be expressed as the latent feature at time t^*.

The third learning objective involves ensuring temporal consistency and predicting future motion matrices. Specifically, the prediction process is formulated as using the sequence of observed motion matrices up to the last observed R-th frame to predict future motion matrices autoregressively:

$$\hat{\mathcal{M}}_{\mathbf{z}_t^*}^{R,R+1} = g_{\phi_4}(\mathcal{M}_{\mathbf{z}_t^*}^{1,2}, \mathcal{M}_{\mathbf{z}_t^*}^{2,3}, \ldots, \mathcal{M}_{\mathbf{z}_t^*}^{R-1,R}), \tag{9}$$

where ϕ_4 represents the parameters of the motion prediction module, here we use ConvLSTM [72] to implement the motion prediction module. The prediction objective is the discrepancy between the predicted motion matrix and the ground truth surrogate motion matrix:

$$\mathcal{L}_{\mathbf{z}}^{\text{motion}} = \sum_{j,t} ||\hat{\mathcal{M}}_{\mathbf{z}_t^*}^{R,R+1} - \mathcal{M}_{\mathbf{z}_t^*}^{R,R+1}||_1. \tag{10}$$

The prediction process helps maintain temporal consistency in the motion information and plays a vital role in enabling the generation of subsequent video frames with only a few base frames.

Despite the potential for error accumulation in autoregressive generation, particularly when dealing with the complexities of motion matrices, we recognize that motion information serves as a more compact representation compared to visual features. For instance, while a moving car exhibits high visual complexity, its motion can be effectively represented by a simple directional vector. We leverage this principle in our motion transformation matrix, condensing motion into a more learnable and concise sequence. This capability enables us to learn and model long-term motion transformations with greater accuracy.

Based on the preceding discussion, the motion learning objective integrates the above three loss terms as follows:

$$\mathcal{L}^{\text{Trans}} = \alpha\mathcal{L}_{\delta\mathbf{z}}^{\text{visual}} + \beta\mathcal{L}_{\mathbf{z}}^{\text{visual}} + \gamma\mathcal{L}_{\mathbf{z}}^{\text{motion}}. \tag{11}$$

Here α, β and γ are the weights for the respective loss components. Through extensive experiment, we determined the optimal weight combination for the loss function and set the hyperparameters as $\alpha = 0.9$, $\beta = 1$, and $\gamma = 0.05$. For more on how these weights impact performance, please see the details in Section V-D.

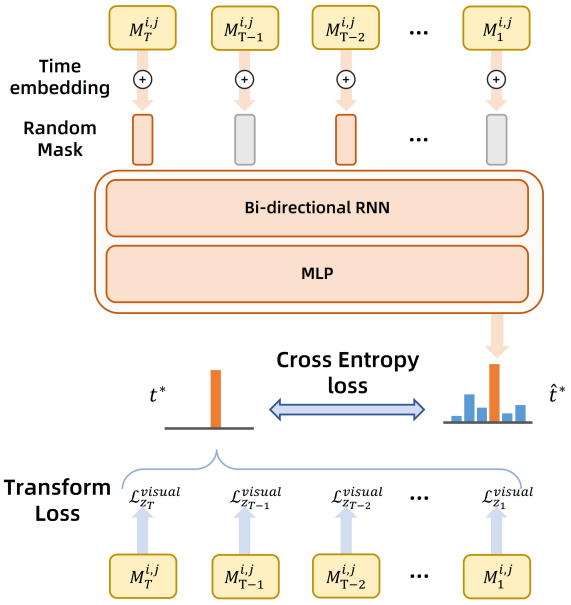


Fig. 6. The DSS module consists of a bidirectional RNN network and an MLP, designed to predict the optimal denoising timestep $\hat{t}$ for transformation based on the input motion matrices at different timesteps. The model is trained using cross-entropy loss between $\hat{t}$ and the ground truth timestep t^*.

C. Denoising Step Selector

DSS is a meta-network designed to predict t^*, the proper intermediate step for switching from motion-based propagations to denoising. The model structure and computation process are illustrated in Figure 6. Specifically, the transition step t^* is determined to be the timestep which leads to the minimal weighted transformation error $\log(\beta \cdot t)\mathcal{L}_{\mathbf{z}_t}^{\text{visual}}$, that is:

$$t^* = \text{argmin}_{t\in\{1,\ldots,T\}} \log(\beta \cdot t) \cdot \mathcal{L}_{\mathbf{z}_t}^{\text{visual}}, \tag{12}$$

where $\mathcal{L}_{\mathbf{z}}^{\text{visual}}$ represents the transformation loss at timestep t, and β is a hyperparameter balancing computational efficiency and transformation quality. Higher values of β prioritize earlier denoising steps to enhance computation efficiency, whereas lower values focus on quality-preserving. However, it is important to note that, given the hierarchical stochastic process in diffusion which generates multi-scale features, this trade-off between efficiency and quality is only effective within a certain range of timesteps. If the timestep is too large or too small, it will lead to incorrect transformations of the features, thereby affecting the generation quality.

To learn t^*, DSS takes motion matrices $\{\mathbf{M}_{\delta\mathbf{z}_t}^{i,j}\}_{t=1}^{T}$ as input, including corresponding timestep indices. It then implements a bi-directional recurrent neural network [73] and outputs $\hat{t}$, the estimated most suitable transition step. DSS is updated according to the cross-entropy loss between the predicted transition step $\hat{t}$ and the ground truth t^*.

During the training process, to compute the ground truth t^* efficiently, we first assume that the motion magnitude between adjacent frames of the same video remains relatively constant, allowing us to calculate t^* only once per video. Furthermore, the relationship between t^* and the transformation loss $\mathcal{L}_{\mathbf{z}_t}^{\text{visual}}$ approximates a quadratic function, as illustrated in Figure 2

TABLE I
RESOURCE USAGE AND PROCESSING TIME FOR DIFFERENT VIDEO PROCESSING SCENARIOS.

	1 timestep	1 video	Filted WebVid (1M video)
GPU	1 A100	1 A100	8 A100
Batch Size	1	1	64
wall-clock time	0.10s	1.13s	4.92h

(right). This relationship enables a heuristic and efficient search for t^*, thus avoiding the need to thoroughly explore all 1000 timesteps of the DDPM sampler. In our method, we only require calculations at about 11.74 timesteps on average to approximate the ground truth t^*. Table I provides a detailed breakdown of the computation times under various conditions.

In addition, we apply a random mask to the input data during training to simulate the case of incomplete information. This strategy ensures that during inference, DSS does not require evaluation of the full sequence but can effectively optimize t^* by analyzing only a subset of the available data, thereby reducing computational demands and speeding up the denoising process.

D. Inference Stage

Our method can generate videos efficiently based on base frames. The base frames can be generated by Diffusion backbone or specified by the user. They serve as inputs to our model, providing initial visual features and initial motion matrices.

During inference, we use DDPM sampler with 1000 timestep and uniformly sampling 10 timesteps. For each sampled timestep, we extract the internal latents of base frame I^0 and I^1 from the U-Net, which we called latent residuals $\delta\mathbf{z}_t^0$ and $\delta\mathbf{z}_t^1$. The Motion Transformation Network (MTN) is then used to compute the motion matrix $\mathbf{M}_{\delta\mathbf{z}_t}^{0,1}$ between $\delta\mathbf{z}_t^0$ and $\delta\mathbf{z}_t^1$ at each timestep t.

Once the sequence of motion matrices $\{\mathbf{M}_{\delta\mathbf{z}_t}^{0,1}\}_{t=1}^{T}$ is obtained, it will be inputted into the Denoising Step Selector (DSS) to predict the optimal transition step t^* of the video. Then the motion matrix sequence $\{\mathbf{M}_{\delta\mathbf{z}_t}^{0,1}\}_{t=t^*}^{T}$ is fed into g_{ϕ_3} to derive the surrogate matrix $\mathcal{M}_{\mathbf{z}_{t^*}}^{0,1}$. This matrix encapsulates the motion between the base frames.

Base on initial $\mathcal{M}_{\mathbf{z}_{t^*}}^{0,1}$, we use g_{ϕ_4} to autoregressively predict the surrogate matrices $\{\mathcal{M}_{\mathbf{z}_{t^*}}^{i,i+1}\}_{i=1}^{L}$ for the next frames. The surrogate matrix is then multiplied with the latent $\mathbf{z}_{t^*}^i$ of the previous frame to obtain the latent $\mathbf{z}_{t^*}^{i+1}$ for the next frame, thus enabling the prediction of the latent sequence $\{\mathbf{z}_{t^*}^i\}_{i=2}^{L}$.

Finally, the latent sequence $\{\mathbf{z}_{t^*}^i\}_{i=2}^{L}$ undergoes a denoising process by diffusion backbone from t^* to generate the video frame $\{I^i\}_{i=2}^{L}$. This diffusion backbone can be either Stable Diffusion or other Diffusion-based video generation model.

The core of our method lies in the fast estimation of coarse-grained features for reusing denoising timesteps, and this approach offers strong flexibility during inference. In addition to only generating the initial base frames using the diffusion backbone, during inference we can dynamically alternate between using the diffusion backbone and motion

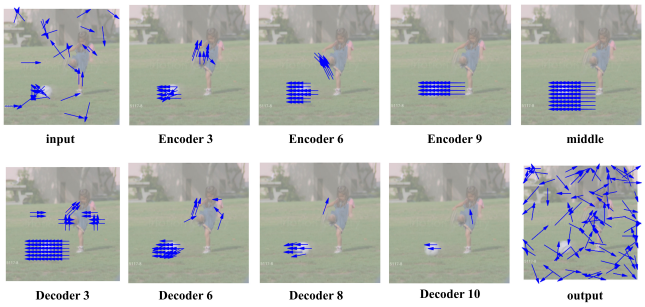


Fig. 7. Visualization of transform matrix from different U-Net blocks.

transformations. For example, after accelerating the generation of K frames with motion transformation, we can use the backbone to generate new frames through the complete denoising process and continue to rapidly predict and reuse features for subsequent frames based on these newly generated frames. This simple yet effective mechanism allows the model to more flexibly leverage the rich visual and motion knowledge in the diffusion backbone, enabling efficient video generation for scenes with complex visual transformations at minimal computational cost.

In addition to the aforementioned DDPM-based inference method, our approach can also be effectively combined with fast samplers, such as DDIM [4]. For instance, when using a DDIMsampler with 50 steps, if the DSS predicts t^* as 270, we round it to the nearest available step(either 260 or 280)to balance speed and visual quality. Specifically, rounding to 260 may offer faster inference, while rounding to 280 could enhance visual quality. In practice, we typically opt for 260 in such cases. This integration with fast samplers further enhances the efficiency of video generation.

V. EXPERIMENTS

This section assesses Dr. Mo's effectiveness in video generation and video editing. Additionally, we conduct ablation studies to investigate the effects of different denoising reuse strategies and varying weights of the loss components, aiming to identify the factors that contribute to the efficacy of our method.

A. Implementation Details

We use Stable Diffusion V1.5 [5] as the backbone, and train the proposed Dr. Mo module on a filtered subset of the WebVid-10M dataset [79]. We perform image resizing and center cropping to 512×512, and downsample the video to 4 fps to avoid low frame-to-frame variance. Training is conducted on the processed video with 20 consecutive frames randomly selected at a time.

We train our model on 8 A100 GPUs. In the training step, in order to improve training efficiency, we first use low resolution for pre-training, in which we resize and center crop the image to 256×256. To train our model on a high resolution, in which we resize and center crop the image to 512×512.

Representations to Construct Motion Matrix By using the output features of each block of the pre-trained Stable

TABLE II
COMPARISON OF VIDEO GENERATION IN TERMS OF VIDEO QUALITY AND EFFICIENCY.

Model	Full Model Parameters	Fine-tuned Parameters	Speed (s)		UCF-101		MSR-VTT	
			256×256	512×512	FVD↓	IS↑	FID↓	CLIPSIM↑
Latent-Shift [8]	1.53B	0.880B	23.40	-	360.04	**92.72**	15.23	0.2773
Latent-VDM [8]	1.58B	0.920B	28.62	-	358.34	90.74	14.35	0.2756
LVDM [33]	1.16B	1.040B	21.23	-	372.00	-	-	0.2930
AnimateDiff [31]	1.38B	0.322B	23.71	79.03	349.37	82.45	14.03	0.3013
SimDA [11]	1.08B	0.025B	11.20	34.20	401.25	79.81	14.76	0.2945
Lumiere [74]	6.5B	6.5B	-	-	332.49	37.54	-	-
W.A.L.T [75]	419M	419M	-	-	344.5	31.7	-	-
PixelDance [76]	1.5B	1.5B	-	-	339.08	-	-	-
Video LDM [77]	4.2B	2.65B	-	-	550.61	33.45	-	0.2929
CogVideo [78]	9.4B	9.40B	8.91	32.68	626.00	50.46	-	-
Make-A-Video [24]	9.72B	9.720B	-	-	367.23	33.00	13.17	0.3049
Dr. Mo (Ours)	1.31B	0.266B	4.35	15.90	**312.81**	89.63	**12.38**	**0.3056**
Dr. Mo (Ours) w/o DSS	1.25B	0.229B	**3.86**	**14.37**	398.72	87.94	14.63	0.2905

Diffusion V1.5 model, we calculated and visualized the inter-frame transform matrix $\mathbf{M}^{i,j}_{\delta \mathbf{z}_t}[b_k]$ from representations of different blocks. As shown in Figure 7, the results showed that the features from U-Net middle layer could achieve a good transform matrix. Ultimately, we select the coarse-grained layer decoder 6 (downsample 16) and the fine-grained layer decoder 8 (downsample 8), which both of which show low transformation loss.We combined the transformation matrices of these two layers using a learnable MLP network.

B. Text-to-Video Generation

We compared Dr. Mo with several recent works, including Latent-Shift [8] and SimDA [11], using text prompts from the test datasets UCF-101 [6] and MSR-VTT [7] to evaluate zero-shot performance.

For UCF-101, we created a template sentence for each category and used that sentence as a text prompt to generate 16 frames without any fine-tuning, setting the base frames $R = 4$ generated by the backbone. We report Frechet Video Distance (FVD) [10] and Inception Score (IS) [9] [2] on 10,000 samples, ensuring that the generated samples match the category distribution of the dataset.

For MSR-VTT, we report Frechet Inception Distance (FID) [80] and CLIPSIM [81] (the average CLIP similarity between video frames and text) using all 2990 captions from the test set [24].

Additionally, considering the significant difficulty variation among different samples in the dataset, where some videos may contain only static backgrounds and slowly moving objects, while others involve complex scenes and motion changes. To better evaluate the model's performance across different scenarios, we used the Structural Similarity Index (SSIM) [82] between the first and last frames of the video to measure the complexity of content changes. We divided the UCF-101 dataset into two subsets: easy ($SSIM \geq 0.6$) and hard ($SSIM < 0.6$).

For inference time, we calculated the average time for generating 16 frames per video using a 50-step DDIM [4] sampler on single A100 GPU, generating 100 videos. For methods with open-source implementations, including LVDM [33], AnimateDiff [31], SimDA [11], and CogVideo [78], we followed their official code and used consistent conditions (same DDIM steps, generated frame count, and resolution) on the single A100 GPU.

For methods without open-source implementations, we referenced the experimental settings and results from the original papers most similar to this work. Latent-Shift and Latent-VDM [8]: Inference times and evaluation metrics were obtained from the original papers, with inference executed on an A100 GPU using DDPM sampling with 100 steps. Lumiere [74]: We used evaluation metrics on UCF-101 from the original paper, with zero-shot text-to-video generation tested on single V5 TPU. W.A.L.T [75]: We compared text-to-video generation results on UCF-101, excluding the 3B-level model which has significantly larger parameters than Dr. Mo. PixelDance [76]: We compared text-conditioned results on UCF-101, excluding those where the tail frame was specified, for a fairer comparison. Video LDM [77]: We referenced and compared zero-shot text-to-video generation results on both UCF-101 and MSR-VTT. Make-A-Video [24]: We referenced and compared zero-shot text-to-video generation results on both UCF-101 and MSR-VTT.

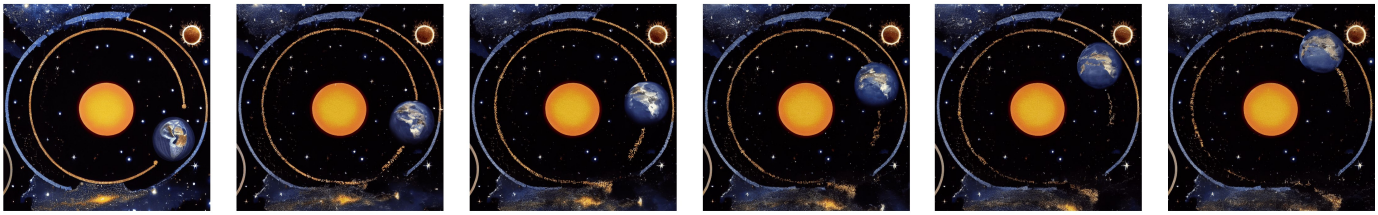
Quantitative Results. As shown in Table II, Dr. Mo outperforms competing video generation models, achieving the lowest FVD score of 312.81 on UCF-101 and the highest CLIPSIM score of 0.3056 on MSR-VTT. These results indicate that Dr. Mo produces videos that closely match real videos in visual and temporal dynamics, and are semantically aligned with their corresponding inputs. The feature reuse in Dr. Mo enhances the consistency of features across frames, reducing the distortion of features that may arise from continuous video changes. Moreover, the video motion learned from coarse-grained features provides more semantically meaningful motion representations, mitigating the confusion between visual feature transformation and motion transformation, which is commonly occurs in diffusion-based video generation and prevents incorrect object deformations. Therefore, Dr. Mo provides superior performance.

Qualitative Results. Figure 8 presents the qualitative results of

(a)A person riding a horse

(b)A minion waved his hand and the UFO flew over

(c)A father and daughter are playing golf



(d)In the universe, the earth revolves around the sun

Fig. 8. Videos generated by Dr. Mo at a resolution of 512×512.

Dr. Mo when generating videos with a resolution of 512×512, demonstrating strong inter-frame consistency and physically plausible motion modeling. Figure 9 shows the results of video generation at a resolution of 256×256 on the UCF-101 dataset, compared against baseline models. More video generation results can be found at our website [1].

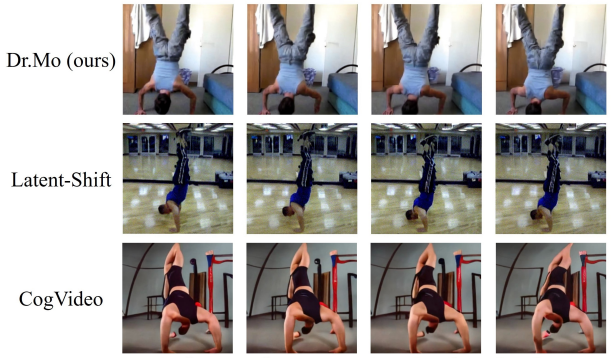
Efficiency Evaluation. As for the computing efficiency, Dr. Mo uses 266M of parameters and achieves the fastest reported inference rates, generating 16×512×512 frames in 15.90 seconds and generating 16×256×256 frames in 4.35 seconds. This is notable considering some current models like Latent-Shift [8] only produce 256×256 resolution images at similar parameter counts. These results suggest that Dr. Mo's design, which optimizes the use of motion information, effectively reduces computational demands and speeds up video generation.

Method Flexibility. Our method is designed based on the characteristics of video data and diffusion models, offering significant flexibility that allows for seamless integration with other diffusion-based video methods to provide acceleration. Specifically, when combined with other diffusion-based video generation backbones, the backbone network first generates at least two base frames to provide initial visual features and extract the initial motion matrices. The Denoising Step Selector (DSS) network then determines the optimal transition step $t = t^*$ for the video, and the Motion Transformation Network (MTN) predicts future motion matrices.These motion matrices are multiplied with the initial visual features to obtain the latent sequence prediction at timestep t^*. Finally, the predicted results are fed into the backbone network for denoising from t^* to 1 to generate the final video output.

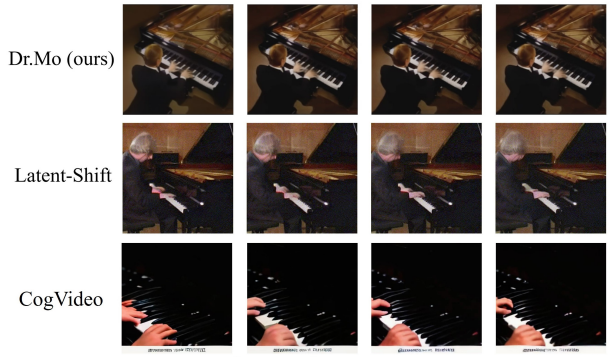
[1]https://drmo-denoising-reuse.github.io/

TABLE III
THE PERFORMANCE EVALUATION OF DR. MO COMBINED WITH OTHER DIFFUSION-BASED VIDEO GENERATION BACKBONES.

	Speed (s)	CLIPSIM↑
AnimateDiff [31]	79.03	0.3079
AnimateDiff with Dr. Mo	28.31	**0.3093**
SimDA [11]	34.20	0.2941
SimDA with Dr. Mo	**17.57**	0.3089



(a) [template]: *A person doing handstand pushups*



(b) [template]: *A person playing piano*

Fig. 9. Comparison with Latent-Shift [8] and CogVideo [78] using video frames with 256×256 resolution on UCF-101.

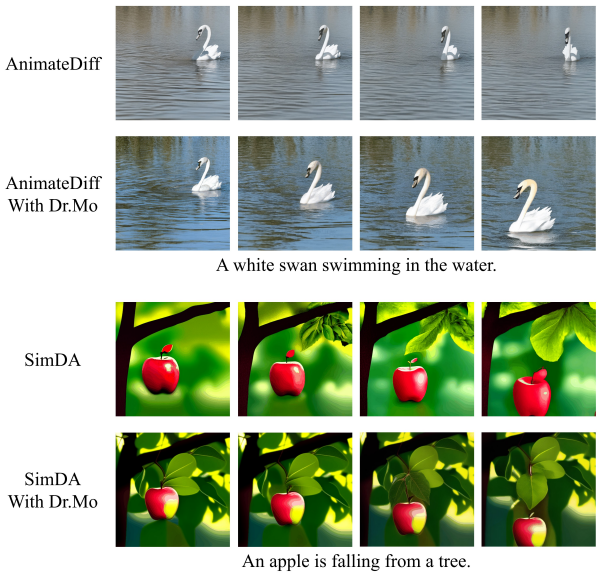


Fig. 10. Comparison of Generation Results with and without Dr.Mo.

In Figure 10, we present results from combining our approach with AnimateDiff [31] and SimDA [11]. The speed metrics are based on the generation of 16-frame videos at a resolution of 512×512. CLIPSIM refers to the average CLIP similarity between the video frames and the prompt. As shown in Table III, integrating Dr. Mo reduces the video generation time by more than half compared to the original backbone, while preserving the original visual style and quality. Furthermore, the reuse of inter-frame features aids in maintaining

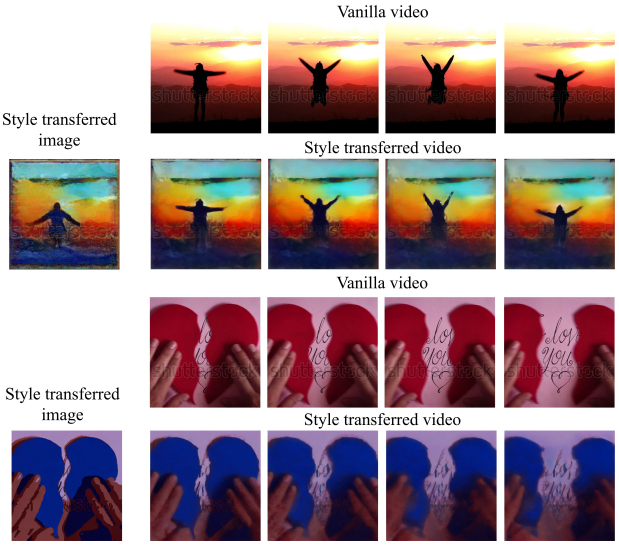


Fig. 11. Video editing Results. By separating motion features, it is possible to apply consistent transformations to different visual features to generate new videos.

feature consistency, mitigating feature collapse over time, and enhancing the semantic coherence of the generated videos. These results highlight the effectiveness and versatility of our method across different frameworks.

C. Video Editing

We evaluate Dr. Mo's video editing capabilities by applying style transformations to real-world videos. Using the motion information from a reference video clip, we extract the motion matrix and apply it to the styled first frame to generate subsequent frames. As shown in Figure 11, Dr. Mo successfully transform real-world videos to match the visual style of the styled base frames. This indicates that the captured motion matrix is disentangled from the visual features and possesses a generalization capability.

D. Ablation Study

Effect of Denoising Reuse. We conduct an ablation study to assess the impact of denoising reuse on video generation performance in Dr. Mo by testing various transition steps at steps 900, 600, 400, 200, and 1. As shown in Figure 12, Dr. Mo performs optimally with 200 denoising steps. This suggests that using the intermediate level of the denoising process as the transition step allows for effective handling of visual features at different granularities in the video.

At step 900, excessive noises mask the motion and visual features lead to ineffective transformations and compromised video content. Conversely, at step 1, the fine-grained visual features being incorrectly transformed lead to accurate overall contours but incorrect appearance details, thereby reducing the video quality.

Effect of DSS Module. Further, we quantitatively examine the effect of DSS by removing it and manually setting different timesteps for transformation. As shown in Table V, Our results show that as t^* decreases, the inference speed improves due to more reused timesteps. However, both excessively large and

TABLE IV
COMPARISON OF VIDEO GENERATION ON EASY AND HARD SUBSETS.

Model	Setting	Speed (s)	Full set		Easy (SSIM≥0.6)		Hard (SSIM<0.6)	
			FVD↓	IS↑	FVD↓	IS↑	FVD↓	IS↑
CogVideo [78]	-	8.91	626.00	50.46	510.46	58.24	804.11	38.46
SimDA [11]	-	11.20	401.25	79.81	356.79	83.05	469.78	74.81
AnimateDiff [31]	-	23.71	349.37	82.45	325.51	84.93	386.15	78.62
Dr. Mo	$R=2$	**3.68**	393.01	87.33	323.09	88.08	500.79	86.17
	$R=3$	4.01	349.28	88.96	318.64	90.18	396.51	87.07
	$R=4$	4.35	312.81	89.63	298.56	**91.02**	334.77	87.48
	$R=6$	5.02	319.90	89.79	300.99	90.34	349.05	88.94
	$R=8$	5.69	310.42	89.84	294.87	91.00	334.39	88.05
	$R=2+2Keyframe$	4.58	**301.08**	**90.08**	**291.44**	90.36	**315.93**	**89.65**

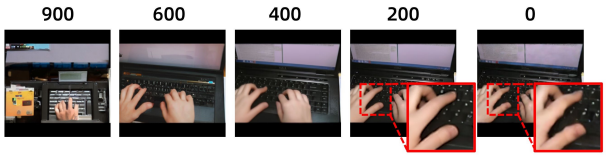


Fig. 12. The result of motion transformation at different t^* values. Too small t^* will produce incorrect appearance details, while too large t^* will lead to the destruction of visual features.

Fig. 13. Left: Example of low motion consistency that requires a larger t^* transformation. Right: Example of high motion consistency that requires a smaller t^* transformation.

small choices of t^* lead to a decrease in generation quality. Optimal visual quality can only be achieved within a suitable range that aligns with the motion features, further emphasizing the importance of dynamically selecting t^* through DSS rather than setting it as a hyperparameter. At the same time, we also tested the heuristic search strategy to select the optimal t^*. This strategy approximates the relationship between the timestep and transform error as a quadratic function, and uses gradient-based binary search to find the optimal timestep approximation. The results show that it has similar quality to the DSS module, but requires an additional 25% of the computation time.

Effect of Varying Motion Consistency. We aim to assess the impact of varying motion consistency on video generation. Following the methodology in MMVP [68], we employ SSIM [82] as a metric and select two data samples with differing consistency from WebVid.

TABLE V
IMPACT OF DSS ON VIDEO GENERATION METRICS. THE FVD AND IS WERE EVALUATED ON THE UCF-101 DATASET. THE SPEED METRICS ARE BASED ON GENERATING 16-FRAME VIDEOS AT A RESOLUTION OF 256×256.

Model	Setting	**FVD↓**	**IS↑**	**Speed (s)**
Lumiere [74]	-	332.49	37.54	-
CogVideo [78]	-	626.00	50.46	8.91
AnimateDiff [31]	-	349.37	82.45	23.71
Dr.Mo	$t^*=1$	485.31	76.02	**2.08**
	$t^*=200$	398.72	87.94	3.86
	$t^*=400$	331.65	85.45	5.25
	$t^*=600$	570.27	69.03	6.93
	$t^*=800$	692.41	57.32	8.76
	$t^*=1000$ (backbone)	819.62	43.51	10.48
	heuristic search	315.43	89.71	5.42
	with DSS	**312.81**	**89.63**	4.35

Fig. 14. Qualitative comparison of different base frame strategies.

The left figure illustrates a video with low motion consistency, with the DSS predicting step 381 as optimal. Our results for steps 381 and 200 show that at step 200, there is a noticeable loss of detail information. Conversely, the right figure shows a video with high motion consistency; here, DSS identifies step 237 as optimal. While the results at step 237 are satisfactory, those at step 400 are less than ideal, due to insufficient learning of motion information. This is attributed to a deficiency in fine-grained visual features and inadequately learned related motion features. These observations highlight the crucial role of motion consistency over time and also validate the effectiveness of the DSS.

Effect of Base Frame. Base frames are crucial as they provide important motion cues and content guidance for the video. We quantitatively examine the impact of the number of Base frames (denoted as R) on the video generation performance. As shown in Table IV, our findings indicate that the model's performance improves as R increases, particularly in complex scenes, due to the richer visual and motion references provided by additional Base frames. However, when $R \geq 4$, the performance gain becomes marginal and stabilizes. Therefore, we set $R = 4$ during inference to achieve better results at a

lower cost. Additionally, we tested an approach that alternates between generating base frames and motion transform. Under the same number of base frames, this method achieves better performance in complex scenes. The visualization results are shown in Figure 14.

TABLE VI
IMPACT OF WEIGHT ADJUSTMENTS ON VIDEO QUALITY METRICS. THE FVD AND IS WERE EVALUATED ON THE UCF-101 DATASET.

α	β	γ	**FVD↓**	**IS↑**
0.0	1.0	0.05	344.38	86.19
2.0	1.0	0.05	325.49	88.00
0.9	0.1	0.05	370.71	81.74
0.9	2.0	0.05	315.40	87.13
0.9	1.0	0.01	332.94	85.99
0.9	1.0	0.50	398.29	79.35
0.9	1.0	0.05	**312.81**	**89.63**

Effect of Loss Components We denote the weights of $\mathcal{L}_{\delta\mathbf{z}}^{\text{visual}}$, $\mathcal{L}_{\mathbf{z}}^{\text{visual}}$, and $\mathcal{L}_{\mathbf{z}}^{\text{motion}}$ as α, β, and γ, respectively. Ablation studies for different weighting configurations are presented in Table VI, where $\alpha = 0.0$ indicates that $\mathcal{L}_{\delta\mathbf{z}}^{\text{visual}}$ loss is not used during training. The results demonstrate that $\mathcal{L}_{\delta\mathbf{z}}^{\text{visual}}$ contributes to the convergence of $\mathcal{L}_{\mathbf{z}}^{\text{visual}}$.This is because enforcing more precise motion matrix extraction on each residual latent improves the overall extraction of motion matrices, leading to more efficient learning. Too high γ negatively impacts the learning, leading to degraded results. This may occur because an excessive emphasis on the predictability of the matrices ($\mathcal{L}_{\mathbf{z}}^{\text{motion}}$) can hinder the matrices be more informative. Ultimately, we chose the weights $\alpha = 0.9$, $\beta = 1.0$ and $\gamma = 0.05$.

VI. CONCLUSION

This paper addresses the efficiency challenges in diffusion-based video generation methods, inspired by a key observation that there is redundancy in inter-frame visual features and that inter-frame motion features remain consistent through most of the diffusion process. The proposed method, called Dr. Mo, enables the reuse of frames across multiple denoising steps, which significantly reduces the need to regenerate each frame from scratch, thereby lowering the computational load and speeding up the video generation process. Frame-specific updates are applied only in the final stages of denoising to maintain the video's integrity and detail. Evaluations of video generation and editing show that our approach provides widely available speedups for diffusion-based video generation models while maintaining backbone visual style and quality.

ACKNOWLEDGMENTS

The computations in this research were performed using the CFFF platform of Fudan University. The work of Yujiang Wang was supported by Basic Research Program of Jiangsu (BK20240414) and SEID Science and Education Leading Talent program (KJQ2024204).

REFERENCES

[1] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[2] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 8633–8646, 2022.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[4] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[5] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[6] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[7] J. Xu, T. Mei, T. Yao, and Y. Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5288–5296.

[8] J. An, S. Zhang, H. Yang, S. Gupta, J.-B. Huang, J. Luo, and X. Yin, "Latent-shift: Latent diffusion with temporal shift for efficient text-to-video generation," *arXiv preprint arXiv:2304.08477*, 2023.

[9] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[10] T. Unterthiner, S. van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Fvd: A new metric for video generation," 2019.

[11] Z. Xing, Q. Dai, H. Hu, Z. Wu, and Y.-G. Jiang, "Simda: Simple diffusion adapter for efficient video generation," *arXiv preprint arXiv:2308.09710*, 2023.

[12] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.

[13] S. Gu, D. Chen, J. Bao, F. Wen, B. Zhang, D. Chen, L. Yuan, and B. Guo, "Vector quantized diffusion model for text-to-image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 696–10 706.

[14] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," *arXiv preprint arXiv:2307.01952*, 2023.

[15] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," *arXiv*

preprint arXiv:2112.10741, 2021.

[16] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[17] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[18] K. Namekata, A. Sabour, S. Fidler, and S. W. Kim, "Emerdiff: Emerging pixel-level semantic knowledge in diffusion models," *arXiv preprint arXiv:2401.11739*, 2024.

[19] Z. Yang, T. Chu, X. Lin, E. Gao, D. Liu, J. Yang, and C. Wang, "Eliminating contextual prior bias for semantic image editing via dual-cycle diffusion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 2, pp. 1316–1320, 2023.

[20] Y. Xue, L.-M. Po, W.-Y. Yu, H. Wu, X. Xu, K. Li, and Y. Liu, "Self-calibration flow guided denoising diffusion model for human pose transfer," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[21] Y. Que, L. Xiong, W. Wan, X. Xia, and Z. Liu, "Denoising diffusion probabilistic model for face sketch-to-photo synthesis," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[22] H. Liu, L. He, M. Zhang, and F. Li, "Vadiffusion: Compressed domain information guided conditional diffusion for video anomaly detection," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[23] D. Li, Y. Liu, Z. Wang, and J. Yang, "Video rescaling with recurrent diffusion," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[24] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni *et al.*, "Make-a-video: Text-to-video generation without text-video data," *arXiv preprint arXiv:2209.14792*, 2022.

[25] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet *et al.*, "Imagen video: High definition video generation with diffusion models," *arXiv preprint arXiv:2210.02303*, 2022.

[26] R. Villegas, M. Babaeizadeh, P.-J. Kindermans, H. Moraldo, H. Zhang, M. T. Saffar, S. Castro, J. Kunze, and D. Erhan, "Phenaki: Variable length video generation from open domain textual descriptions," in *International Conference on Learning Representations*, 2022.

[27] Y. Wang, P. Bilinski, F. Bremond, and A. Dantcheva, "Imaginator: Conditional spatio-temporal gan for video generation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 1160–1169.

[28] Q. Chen, Q. Wu, J. Chen, Q. Wu, A. van den Hengel, and M. Tan, "Scripted video generation with a bottom-up generative adversarial network," *IEEE Transactions on Image Processing*, vol. 29, pp. 7454–7467, 2020.

[29] S. Gupta, A. Keshari, and S. Das, "Rv-gan: Recurrent gan for unconditional video generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 2024–2033.

[30] P. Esser, J. Chiu, P. Atighehchian, J. Granskog, and A. Germanidis, "Structure and content-guided video synthesis with diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 7346–7356.

[31] Y. Guo, C. Yang, A. Rao, Y. Wang, Y. Qiao, D. Lin, and B. Dai, "Animatediff: Animate your personalized text-to-image diffusion models without specific tuning," *arXiv preprint arXiv:2307.04725*, 2023.

[32] J. Z. Wu, Y. Ge, X. Wang, S. W. Lei, Y. Gu, Y. Shi, W. Hsu, Y. Shan, X. Qie, and M. Z. Shou, "Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 7623–7633.

[33] Y. He, T. Yang, Y. Zhang, Y. Shan, and Q. Chen, "Latent video diffusion models for high-fidelity long video generation," *arXiv preprint arXiv:2211.13221*, 2022.

[34] Y. Wang, X. Chen, X. Ma, S. Zhou, Z. Huang, Y. Wang, C. Yang, Y. He, J. Yu, P. Yang *et al.*, "Lavie: High-quality video generation with cascaded latent diffusion models," *arXiv preprint arXiv:2309.15103*, 2023.

[35] L. Khachatryan, A. Movsisyan, V. Tadevosyan, R. Henschel, Z. Wang, S. Navasardyan, and H. Shi, "Text2video-zero: Text-to-image diffusion models are zero-shot video generators," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 15 954–15 964.

[36] X. Wang, S. Zhang, H. Zhang, Y. Liu, Y. Zhang, C. Gao, and N. Sang, "Videolcm: Video latent consistency model," *arXiv preprint arXiv:2312.09109*, 2023.

[37] J. Gu, S. Wang, H. Zhao, T. Lu, X. Zhang, Z. Wu, S. Xu, W. Zhang, Y.-G. Jiang, and H. Xu, "Reuse and diffuse: Iterative denoising for text-to-video generation," *arXiv preprint arXiv:2309.03549*, 2023.

[38] A. Barjatya, "Block matching algorithms for motion estimation," *IEEE Transactions Evolution Computation*, vol. 8, no. 3, pp. 225–239, 2004.

[39] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[40] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462–2470.

[41] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8934–8943.

[42] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field

transforms for optical flow," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 402–419.

[43] X. Xiang, R. Abdein, and N. Lv, "Unsupervised optical flow estimation method based on transformer and occlusion compensation," *Neural Computing and Applications*, vol. 34, no. 17, pp. 14 341–14 353, 2022.

[44] S. Zhao, Y. Sheng, Y. Dong, E. I. Chang, Y. Xu *et al.*, "Maskflownet: Asymmetric feature matching with learnable occlusion mask," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6278–6287.

[45] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, pp. 91–110, 2004.

[46] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 224–236.

[47] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic, "Neighbourhood consensus networks," *Advances in neural information processing systems*, vol. 31, 2018.

[48] Y. Shi, J.-X. Cai, Y. Shavit, T.-J. Mu, W. Feng, and K. Zhang, "Clustergnn: Cluster-based coarse-to-fine graph neural network for efficient feature matching," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 12 517–12 526.

[49] X. He, J. Sun, Y. Wang, S. Peng, Q. Huang, H. Bao, and X. Zhou, "Detector-free structure from motion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 594–21 603.

[50] B. Chen, Z. Wang, B. Li, S. Wang, and Y. Ye, "Compact temporal trajectory representation for talking face video compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 11, pp. 7009–7023, 2023.

[51] K. Lin, C. Jia, X. Zhang, S. Wang, S. Ma, and W. Gao, "Dmvc: Decomposed motion modeling for learned video compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 7, pp. 3502–3515, 2022.

[52] Z. Chen, T. He, X. Jin, and F. Wu, "Learning for video compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 2, pp. 566–576, 2019.

[53] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, and S. Wang, "Image and video compression with neural networks: A review," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1683–1698, 2019.

[54] Y. Wang, Y. Yang, Z. Yang, L. Zhao, P. Wang, and W. Xu, "Occlusion aware unsupervised learning of optical flow," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4884–4893.

[55] Z. Yang, R. Simon, Y. Li, and C. A. Linte, "Dense depth estimation from stereo endoscopy videos using unsupervised optical flow methods," in *Medical Image Understanding and Analysis: 25th Annual Conference, MIUA 2021, Oxford, United Kingdom, July 12–14, 2021, Proceedings 25*. Springer, 2021, pp. 337–349.

[56] S. Xu, S. Chen, R. Xu, C. Wang, P. Lu, and L. Guo, "Local feature matching using deep learning: A survey," *Information Fusion*, vol. 107, p. 102344, 2024.

[57] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[58] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*. Springer, 2016, pp. 467–483.

[59] K. Fu, Z. Liu, X. Wu, C. Sun, and W. Chen, "An effective end-to-end image matching network with attentional graph neural networks," in *2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2022, pp. 1628–1633.

[60] Y. Tian, V. Balntas, T. Ng, A. Barroso-Laguna, Y. Demiris, and K. Mikolajczyk, "D2d: Keypoint extraction with describe to detect approach," in *Proceedings of the Asian conference on computer vision*, 2020.

[61] K. Li, L. Wang, L. Liu, Q. Ran, K. Xu, and Y. Guo, "Decoupling makes weakly supervised local feature better," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 15 838–15 848.

[62] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4938–4947.

[63] H. Chen, Z. Luo, L. Zhou, Y. Tian, M. Zhen, T. Fang, D. Mckinnon, Y. Tsin, and L. Quan, "Aspanformer: Detector-free image matching with adaptive span transformer," in *European Conference on Computer Vision*. Springer, 2022, pp. 20–36.

[64] W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, and K. M. Yi, "Cotr: Correspondence transformer for matching across images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6207–6217.

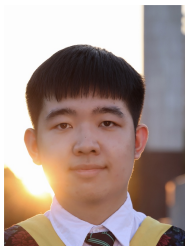
[65] P. Lindenberger, P.-E. Sarlin, V. Larsson, and M. Pollefeys, "Pixel-perfect structure-from-motion with featuremetric refinement," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5987–5997.

[66] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

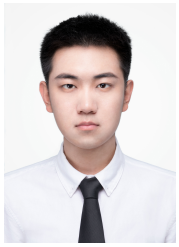
[67] A. Zadaianchuk, M. Seitzer, and G. Martius, "Object-centric learning for real-world videos by predicting temporal feature similarities," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[68] Y. Zhong, L. Liang, I. Zharkov, and U. Neumann, "Mmvp: Motion-matrix-based video prediction," in *Pro-*

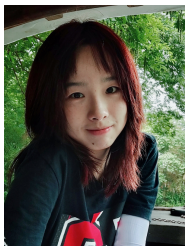
ceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4273–4283.
[69] Y. Kim, D. Jo, H. Jeon, T. Kim, D. Ahn, H. Kim *et al.*, "Leveraging early-stage robustness in diffusion models for efficient and high-quality image synthesis," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
[70] D. Baranchuk, I. Rubachev, A. Voynov, V. Khrulkov, and A. Babenko, "Label-efficient semantic segmentation with diffusion models," *arXiv preprint arXiv:2112.03126*, 2021.
[71] H. Liu, C. Xu, Y. Yang, L. Zeng, and S. He, "Drag your noise: Interactive point-based editing via diffusion semantic propagation," *arXiv preprint arXiv:2404.01050*, 2024.
[72] Y. Wang, L. Jiang, M.-H. Yang, L.-J. Li, M. Long, and L. Fei-Fei, "Eidetic 3d lstm: A model for video prediction and beyond," in *International conference on learning representations*, 2018.
[73] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.
[74] O. Bar-Tal, H. Chefer, O. Tov, C. Herrmann, R. Paiss, S. Zada, A. Ephrat, J. Hur, G. Liu, A. Raj *et al.*, "Lumiere: A space-time diffusion model for video generation," in *SIGGRAPH Asia 2024 Conference Papers*, 2024, pp. 1–11.
[75] A. Gupta, L. Yu, K. Sohn, X. Gu, M. Hahn, F.-F. Li, I. Essa, L. Jiang, and J. Lezama, "Photorealistic video generation with diffusion models," in *European Conference on Computer Vision*. Springer, 2025, pp. 393–411.
[76] Y. Zeng, G. Wei, J. Zheng, J. Zou, Y. Wei, Y. Zhang, and H. Li, "Make pixels dance: High-dynamic video generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 8850–8860.
[77] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis, "Align your latents: High-resolution video synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 563–22 575.
[78] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang, "Cogvideo: Large-scale pretraining for text-to-video generation via transformers," *arXiv preprint arXiv:2205.15868*, 2022.
[79] M. Bain, A. Nagrani, G. Varol, and A. Zisserman, "Frozen in time: A joint video and image encoder for end-to-end retrieval," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1728–1738.
[80] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.
[81] C. Wu, L. Huang, Q. Zhang, B. Li, L. Ji, F. Yang, G. Sapiro, and N. Duan, "Godiva: Generating open-domain videos from natural descriptions," *arXiv preprint arXiv:2104.14806*, 2021.
[82] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.



Chenyu Wang received the B.Eng. degree in intelligent science and technology from Xidian University, in 2022. His research interests include deep learning, computer vision, and video generation.



Shuo Yan received the B.Eng. degree in digital media technology from Shandong University, in 2021. His research interests include deep learning, computer vision, and video generation.



Yixuan Chen Yixuan Chen received the Ph.D. degree in computer science from Fudan University Shanghai, China, in 2024, and the M.S. degree in management science and engineering from the University of Chinese Academy of Sciences, Beijing, in 2020. Her research interests include vision-language models and machine learning in healthcare.

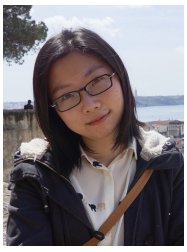
Xianwei Wang received the B.Eng. degree in software engineering from Xiamen University, in 2022. His research interests include deep learning, computer vision, and Image generation.

Yujiang Wang is a Research Scientist and Associate Head of Machine Learning Lab at Oxford Suzhou Centre for Advanced Research, Suzhou, China. He received his PhD degree in computer vision from Imperial College London, and his BSc and two honorary MSc degrees from Tsinghua University, Imperial College London, and University College London, respectively. His research interests focus on medical AI, lip recognition, and smart wearables.

Mingzhi Dong is a Lecturer at the School of Computing, University of Bath, UK, and a PhD at University College London, UK. His research area covers theoretical or intuitively interpretable machine learning algorithms and their applications in computer vision, human-computer interaction, and other fields.

Xiaochen Yang is a Lecturer in the School of Mathematics and Statistics at the University of Glasgow, UK, and a PhD in Statistical Science at University College London. Her research interests cover small sample machine learning, credible machine learning, and machine learning methods in medical sciences.

Dongsheng Li (Member, IEEE) is a Principal Research Manager at Microsoft Research Asia (MSRA) Shanghai, an Adjunct Professor at the School of Computer Science, Fudan University, a former researcher at IBM China Research Shanghai, a Senior Member of ACM and IEEE, and a member of the program committee of several top conferences, such as NIPS, ICML, ICLR, etc. He is also a member of the Program Committee of Microsoft Research Asia (MSRA) Shanghai, an Adjunct Professor at the School of Computer Science, Fudan University. His research interests include machine learning, health AI, and recommender systems.



Rui Zhu is an Associate Professor at the Bayesian Business School, City College, University of London, and has a PhD from the Department of Statistical Science, University College London. Her research interests include statistical learning, subspace-based classification methods, spectral data analysis, hyperspectral image analysis, and image quality evaluation.

David Clifton is the Royal Academy of Engineering Chair of Clinical Machine Learning at the University of Oxford, where he leads the Computational Health Informatics (CHI) Lab. He holds NIHR Research Professor, Fellow of the Alan Turing Institute, and Visiting Chair in AI for Health at the University of Manchester. He has received over 40 awards, including the IEEE Early Career Award in 2022.His research focuses on AI for healthcare, particularly in the development of AI-driven systems used in both hospital and home settings.

Robert P. Dick (Senior Member, IEEE) is a professor in the Department of Electrical Engineering and Computer Science, College of Engineering, University of Michigan, and a Ph.D. in Electrical Engineering at Princeton University. He was an associate professor in the Department of Electrical Engineering and Computer Science at Northwestern University and a visiting professor in the Department of Electrical Engineering at Tsinghua University. His research areas cover embedded systems, high efficiency, bionic machine learning, learning dynamics, privacy protection, and adversarial censorship.

Qin Lv is a professor in the Department of Computer Science at the University of Colorado at Boulder and has a PhD in Computer Science from Princeton University. Research focuses on full-stack data analytics, integrating systems, algorithms, and applications for efficient and effective data analysis in pervasive computing and scientific discovery.

Fan Yang is a professor and a doctoral director at School of Microelectronics, Fudan University, and a Ph.D. in microelectronics and solid state electronics at Fudan University. His research areas cover acceleration of artificial neural networks, integrated circuit yield analysis methods and optimization, high-level synthesis methods, fast analytical modeling of large-scale circuits, and parallel computing methods.

Tun Lu (Member, IEEE) is a professor and PhD supervisor at the School of Computer Science and Technology, Fudan University, and vice dean of the School of Computer Science and Technology, Fudan University. His research areas cover CSCW and Social Computing, Human-Computer Collaboration and Interaction, Group Intelligence Collaboration, Recommender Systems, Service and Cloud Computing.

Ning Gu (Member, IEEE) is a professor and doctoral supervisor at the School of Computer Science and Technology, Fudan University, director of the Social Computing Research Center at Fudan University, Fellow of the Chinese Computer Society, Fellow of the IET (Institution of Engineering and Technology, UK), and Honorary Director of the CCF Special Committee on Collaborative Computing. He has long been engaged in research on human-centered collaborative computing, including the theory and technology of distributed collaboration, social collaboration and group intelligence collaboration.

Li Shang (Member, IEEE) is the Shanghai specially-appointed expert, PhD from Princeton University, a former vice president and chief architect of Intel China Research Institute, and received a (tenure-track) faculty position from the University of Colorado at Boulder, USA. Focusing on the development of human-centered intelligent AGI systems, AI computing systems, and intelligences, his research results have been published in more than 170 papers in top international conferences and journals such as ICLR, NeurIPS, WWW, IMWUT, Nature Communication, etc., and he has been awarded and nominated for best paper awards several times, and has been cited more than 8,000 times. He also has rich experience in industry.