Conditional Image-to-Video Generation with Latent Flow Diffusion Models

Haomiao Ni[1*]   Changhao Shi[2*]   Kai Li[3]   Sharon X. Huang[1]   Martin Renqiang Min[3]

[1]The Pennsylvania State University, University Park, PA, USA
[2]University of California, San Diego, CA, USA
[3]NEC Laboratories America, Princeton, NJ, USA

[1]{hfn5052, suh972}@psu.edu   [2]cshi@ucsd.edu   [3]{kaili, renqiang}@nec-labs.com

Abstract

Conditional image-to-video (cI2V) generation aims to synthesize a new plausible video starting from an image (e.g., a person's face) and a condition (e.g., an action class label like smile). The key challenge of the cI2V task lies in the simultaneous generation of realistic spatial appearance and temporal dynamics corresponding to the given image and condition. In this paper, we propose an approach for cI2V using novel latent flow diffusion models (LFDM) that synthesize an optical flow sequence in the latent space based on the given condition to warp the given image. Compared to previous direct-synthesis-based works, our proposed LFDM can better synthesize spatial details and temporal motion by fully utilizing the spatial content of the given image and warping it in the latent space according to the generated temporally-coherent flow. The training of LFDM consists of two separate stages: (1) an unsupervised learning stage to train a latent flow auto-encoder for spatial content generation, including a flow predictor to estimate latent flow between pairs of video frames, and (2) a conditional learning stage to train a 3D-UNet-based diffusion model (DM) for temporal latent flow generation. Unlike previous DMs operating in pixel space or latent feature space that couples spatial and temporal information, the DM in our LFDM only needs to learn a low-dimensional latent flow space for motion generation, thus being more computationally efficient. We conduct comprehensive experiments on multiple datasets, where LFDM consistently outperforms prior arts. Furthermore, we show that LFDM can be easily adapted to new domains by simply finetuning the image decoder. Our code is available at https://github.com/nihaomiao/CVPR23_LFDM.

1. Introduction

Image-to-video (I2V) generation is an appealing topic and has many potential applications, such as artistic cre-
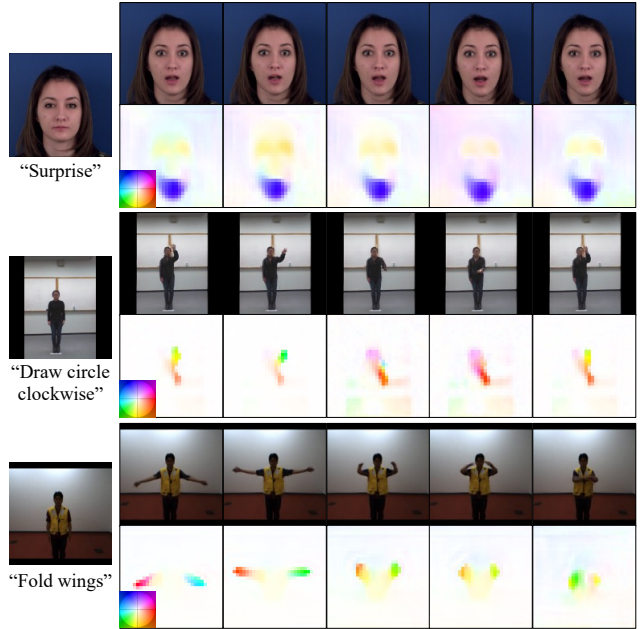
Figure 1. Examples of generated video frames and latent flow sequences using our proposed LFDM. The first column shows the given images x_0 and conditions y. The latent flow maps are *backward* optical flow to x_0 in the *latent* space. We use the color coding scheme in [4] to visualize flow, where the color indicates the direction and magnitude of the flow.

ation, entertainment and data augmentation for machine learning [30]. Given a single image x_0 and a condition y, conditional image-to-video (cI2V) generation aims to synthesize a realistic video with frames 0 to k, $\hat{\mathbf{x}}_0^K = \{x_0, \hat{x}_1, \ldots, \hat{x}_K\}$, starting from the given frame x_0 and satisfying the condition y. Similar to conditional image synthesis works [43, 82], most existing cI2V generation methods [16, 19, 21, 30, 36, 77] directly synthesize each frame in the whole video based on the given image x_0 and condition y. However, they often struggle with simultaneously preserving spatial details and keeping temporal coherence in the generated frames. In this paper, we propose novel latent
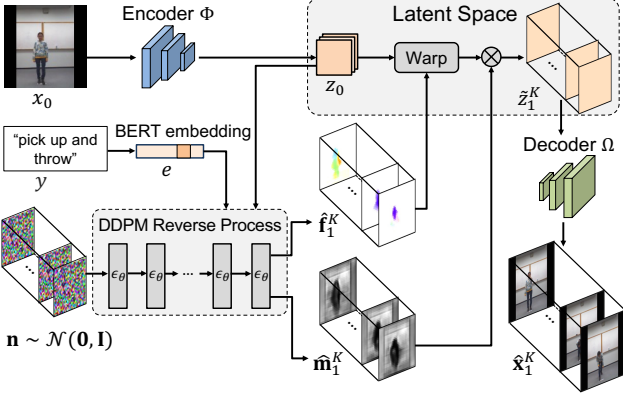
Figure 2. The video generation (*i.e.*, inference) process of LFDM. The generated flow sequence $\hat{\mathbf{f}}_1^K$ and occlusion map sequence $\hat{\mathbf{m}}_1^K$ have the same *spatial* size as image latent map z_0. The brighter regions in $\hat{\mathbf{m}}_1^K$ mean those are regions less likely to be occluded.

flow diffusion models, termed *LFDM*, which mitigate this issue by synthesizing a latent optical flow sequence conditioned on y, to warp the image x_0 in the *latent* space for generating new videos (see Fig. 1 for an example). Unlike previous direct-synthesis or warp-free based methods, the spatial content of the given image can be consistently reused by our warp-based LFDM through the generated temporally-coherent flow. So LFDM can better preserve subject appearance, ensure motion continuity and also generalize to unseen images, as shown in Sec. 4.3.

To disentangle the generation of spatial content and temporal dynamics, the training of LFDM is designed to include two separate stages. In stage one, inspired by recent motion transfer works [50, 62, 75, 76, 79], a latent flow auto-encoder (LFAE) is trained in an unsupervised fashion. It first estimates the latent optical flow between two frames from the same video, a reference frame and a driving frame. Then the reference frame is warped with predicted flow and LFAE is trained by minimizing the reconstruction loss between this warped frame and the driving frame. In stage two, a 3D U-Net [12] based diffusion model (DM) is trained using paired condition y and latent flow sequence extracted from training videos using trained LFAE. Conditioned on y and image x_0, the DM learns how to produce temporally coherent latent flow sequences through 3D convolutions. Different from previous DMs learning in a high-dimensional pixel space [20, 24, 27, 65, 73, 84] or latent feature space [57] that couples spatial and temporal information, the DM in LFDM operates in a *simple* and *low-dimensional* latent flow space which only describes motion dynamics. So the diffusion generation process can be more computationally efficient. Benefiting from the decoupled training strategy, LFDM can be easily adapted to new domains. In Sec. 4.3, we show that using the temporal latent flow produced by DM trained in the original domain, LFDM can animate fa-

cial images from a new domain and generate better spatial details if the image decoder is finetuned.

During inference, as Fig 2 shows, we first adopt the DM trained in stage two to generate a latent flow sequence $\hat{\mathbf{f}}_1^K$ conditioned on y and given image x_0. To generate the occluded regions in new frames, the DM also produces an occlusion map sequence $\hat{\mathbf{m}}_1^K$. Then image x_0 is warped with $\hat{\mathbf{f}}_1^K$ and $\hat{\mathbf{m}}_1^K$ in the latent space frame-by-frame to generate the video $\hat{\mathbf{x}}_1^K$. By keeping warping the given image x_0 instead of previous synthesized frames, we can avoid artifact accumulation. More details will be introduced in Sec. 3.3.

Our contributions are summarized as follows:

- We propose novel latent flow diffusion models (LFDM) to achieve image-to-video generation by synthesizing a temporally-coherent flow sequence in the latent space based on the given condition to warp the given image. To the best of our knowledge, we are the first to apply diffusion models to generate latent flow for conditional image-to-video tasks.

- A novel two-stage training strategy is proposed for LFDM to decouple the generation of spatial content and temporal dynamics, which includes training a latent flow auto-encoder in stage one and a conditional 3D U-Net based diffusion model in stage two. This disentangled training process also enables LFDM to be easily adapted to new domains.

- We conduct extensive experiments on multiple datasets, including videos of facial expression, human action and gesture, where our proposed LFDM consistently outperforms previous state-of-the-art methods.

2. Related Work

2.1. Image-to-Video Generation

Conditional video generation aims to synthesize videos guided by user-provided signals. According to which kind of conditions are provided, conditional video generation can be categorized into text-to-video (T2V) generation [5, 13, 29, 42, 48, 49, 54, 80], video-to-video (V2V) generation [10, 47, 50, 75, 76, 78], and image-to-video (I2V) generation [7, 8, 16, 30, 46, 53, 77, 81, 83, 85]. I2V is also closely related to video prediction from single images [3, 36, 41]. Depending on the availability of motion cues, I2V can be further classified into stochastic (*i.e.,* only using given image x_0) [3, 41, 53, 81, 85] and conditional generation [7, 8, 16, 30, 36, 46, 77, 83] (using x_0 and given condition y). Here we mainly discuss previous conditional image-to-video (cI2V) generation methods.

Yang *et al*. [83] proposed a pose-guided method by first extracting pose from given image x_0 and learning a GAN model to produce pose sequence conditioned on y. Then another GAN model was employed to synthesize video frames

2

from the poses. However, direct pose-to-video synthesis may be difficult to produce fine-grained details. Blattmann *et al.* [7, 8] proposed interactive I2V models which allowed users to specify the desired motion through manual poking of a pixel. Mahapatra *et al.* [46] achieved cI2V by estimating optical flow maps from motion direction inputs. Though these methods do not require users to provide detailed guidance, it is hard for them to generate videos with complex motions. Wang *et al.* [77] proposed a GAN model ImaGINator, which includes a specially designed spatio-temporal fusion scheme and transposed (1+2)D convolution. Hu *et al.* [30] proposed a I2V generator with a motion anchor structure to store appearance-motion aligned representations. However, these direct synthesis methods require modeling both spatial and temporal features, which may complicate the network training.

Differences from previous flow-based I2V works. Several stochastic I2V works [41, 53, 85] designed warping-flow-based models, which generated flow using GAN [17] or VAE [39] and only conditioned on given image x_0. Our LFDM instead generates flow sequences based on both image x_0 and condition y using diffusion models [25], which have emerged as a new paradigm in generation tasks. Moreover, unlike their separately-trained flow predictor and flow-warped frame generator, the latent flow auto-encoder in LFDM learns these modules together in an unsupervised fashion. Although the cI2V method in [46] also adopted warping flow to model the motion of fluid elements (water, smoke, fire, *etc.*), their flow maps can be easily acquired from motion direction inputs. This is because fluid elements can follow the assumption of time-invariant motion field [28], which does not hold for complex motions such as human movements.

2.2. Diffusion Models for Video Generation

Diffusion models (DMs) [25, 67] have very recently found remarkable success in image generation [2, 15, 18, 26, 32, 35, 44, 51, 52, 55–57, 60, 61]. For example, Rombach [57] proposed latent diffusion models (LDM) for image generation by applying DM in the latent space of pretrained auto-encoders. For video generation, DMs also showed promising performance [20, 24, 27, 65, 74, 84]. Ho *et al.* [27] introduced video diffusion models (VDM) by changing the 2D U-Net [58] in image DM to be 3D U-Net [12]. In [24], they further utilized VDM to design a base video generation model and improve it with a sequence of video super-resolution models. Singer *et al.* [65] exploited pretrained text-to-image DMs to achieve text-to-video generation without paired text-video data. Different from all the above models, LFDM instead applies DM to generate latent flow sequences for conditional image-to-video generation.

3. Our Method

Let $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ be a Gaussian noise volume with the shape of $K_n \times H_n \times W_n \times C_n$, where K_n H_n, W_n, and C_n are length, height, width, and channel number, respectively. Given one starting image x_0 and condition y, let $\mathbf{x}_0^K = \{x_0, x_1, \ldots, x_K\}$ be the real video of condition y, the goal of conditional image-to-video (cI2V) generation is to learn a mapping that converts the noise volume $\mathbf{n}$ to a synthesized video, $\hat{\mathbf{x}}_1^K = \{\hat{x}_1, \ldots, \hat{x}_K\}$, so that the conditional distribution of $\hat{\mathbf{x}}_1^K$ given x_0 and y is identical to the conditional distribution of $\mathbf{x}_1^K$ given x_0 and y, *i.e.*, $p(\hat{\mathbf{x}}_1^K|x_0, y) = p(\mathbf{x}_1^K|x_0, y)$. Similar to [36, 77, 83], we only consider the class label as the input condition y. In this section, we first introduce the background of diffusion models, then explain our proposed two-stage training framework in LFDM and finally illustrate our inference process.

3.1. Diffusion Models

Our proposed LFDM is built on denoising diffusion probabilistic models (DDPM) [25, 67, 70]. Given a sample from the data distribution $\mathbf{s}_0 \sim q(\mathbf{s}_0)$, the *forward* process of DDPM produces a Markov chain $\mathbf{s}_1, \ldots, \mathbf{s}_T$ by progressively adding Gaussian noise to $\mathbf{s}_0$ according to a variance schedule $\beta_1, \ldots, \beta_T$, that is:

$$q(\mathbf{s}_t|\mathbf{s}_{t-1}) = \mathcal{N}(\mathbf{s}_t; \sqrt{1-\beta_t}\mathbf{s}_{t-1}, \beta_t\mathbf{I}) \ , \qquad (1)$$

where variances β_t are held constant. When β_t are small, the posterior $q(\mathbf{s}_{t-1}|\mathbf{s}_t)$ can be well approximated by diagonal Gaussian [51, 67]. Moreover, if the T of the chain is large enough, $\mathbf{s}_T$ can be well approximated by standard Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$. These suggest that the true posterior $q(\mathbf{s}_{t-1}|\mathbf{s}_t)$ can be estimated by $p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t)$ defined as:

$$p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t) = \mathcal{N}(\mathbf{s}_{t-1}; \mu_\theta(\mathbf{s}_t), \sigma_t^2\mathbf{I}) \ , \qquad (2)$$

where variances σ_t are also constants. The DDPM *reverse* process (also termed *sampling*) then produces samples $\mathbf{s}_0 \sim p_\theta(\mathbf{s}_0)$ by starting with Gaussian noise $\mathbf{s}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and gradually reducing noise in a Markov chain of $\mathbf{s}_{T-1}, \mathbf{s}_{T-2}, \ldots, \mathbf{s}_0$ with learnt $p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t)$. To learn $p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t)$, Gaussian noise ϵ is added to $\mathbf{s}_0$ to generate samples $\mathbf{s}_t \sim q(\mathbf{s}_t|\mathbf{s}_0)$, then a model ϵ_θ is trained to predict ϵ using the following mean-squared error loss:

$$L = \mathbb{E}_{t\sim\mathcal{U}(1,T),\mathbf{s}_0\sim q(\mathbf{s}_0),\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}[||\epsilon - \epsilon_\theta(\mathbf{s}_t, t)||^2] \ , \quad (3)$$

where time step t is uniformly sampled from $\{1, \ldots, T\}$. Then $\mu_\theta(\mathbf{s}_t)$ in Eq. 2 can be derived from $\epsilon_\theta(\mathbf{s}_t, t)$ to model $p_\theta(\mathbf{s}_{t-1}|\mathbf{s}_t)$ [25]. Denoising model ϵ_θ is usually implemented via a time-conditioned U-Net [58] with residual blocks [22] and self-attention layers [72]. And time step t is specified to ϵ_θ by the sinusoidal position embedding [72]. For conditional generation, *i.e.*, sampling $\mathbf{s}_0 \sim p_\theta(\mathbf{s}_0|y)$, one can simply learn a y-conditioned model $\epsilon_\theta(\mathbf{s}_t, t, y)$ [51, 57]. Recently, Ho *et al.* [26] proposed *classifier-free guidance* for conditional generation in DMs. During training, the condition y in $\epsilon_\theta(\mathbf{s}_t, t, y)$ is replaced by a null label
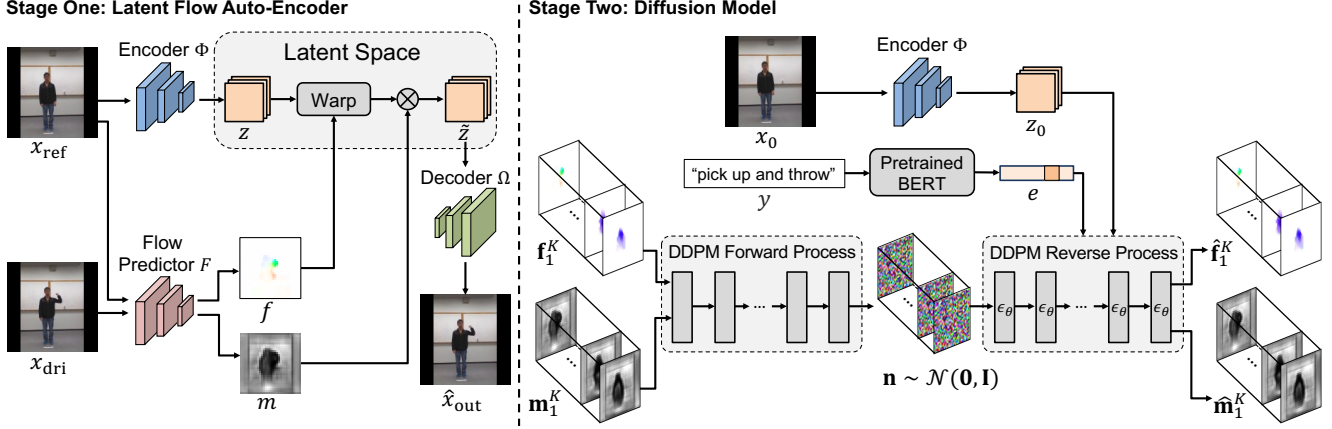
Figure 3. The training framework of LFDM. On the left is stage one for training latent flow auto-encoder while on the right is stage two for training diffusion model. In stage two, the encoder Φ is the one already trained in stage one, and the latent flow sequence $\mathbf{f}_1^K$ and occlusion map sequence $\mathbf{m}_1^K$ are estimated between x_0 and each frame in ground truth video $\mathbf{x}_1^K$ using the trained flow predictor F from stage one.

$\emptyset$ with a fixed probability. During sampling, the model output is generated as follows:

$$\hat{\epsilon}_\theta(\mathbf{s}_t, t, y) = \epsilon_\theta(\mathbf{s}_t, t, \emptyset) + g \cdot (\epsilon_\theta(\mathbf{s}_t, t, y) - \epsilon_\theta(\mathbf{s}_t, t, \emptyset)) \ , \quad (4)$$

where g is the guidance scale.

3.2. Training

The overall training process of LFDM is shown in Fig. 3, which includes two separate stages to decouple the generation of spatial content and temporal dynamics. In stage one, a latent flow auto-encoder (LFAE) is trained in an unsupervised fashion to estimate latent flow between a pair of video frames, a reference frame and a driving frame, and the reference frame is warped with the latent flow to reconstruct the driving frame. In stage two, we train a 3D-UNet-based diffusion model (DM) to produce temporally-coherent latent flow sequence conditioned on image x_0 and class y. Implementation details will be presented in Sec 4.2.

3.2.1 Stage One: Latent Flow Auto-Encoder

In stage one, we train a latent flow auto-encoder (LFAE) in an unsupervised manner. As Fig. 3 shows, LFAE contains three trainable modules: an image encoder Φ to represent image x as latent map z, a flow predictor F to estimate the latent flow f and occlusion map m between video frames, and an image decoder Ω to decode warped latent map $\tilde{z}$ as output image $\hat{x}$. During the stage-one training, we first randomly select two frames from the *same* video, a reference frame x_{ref} and a driving frame x_{dri}. Both x_{ref} and x_{dri} are RGB frames so their size is $H_x \times W_x \times 3$. Encoder Φ then encodes x_{ref} as latent map z with the size of $H_z \times W_z \times C_z$. x_{ref} and x_{dri} are also fed to flow predictor F to estimate the *backward* latent flow f from x_{dri} to x_{ref}, which has the same *spatial* size ($H_z \times W_z \times 2$) as latent map z. Here flow f has 2 channels because it describes the horizontal and vertical movement between frames. And we employ *backward*

flow because it can be efficiently implemented through a differentiable bilinear sampling operation [31]. However, only using f to warp latent z may be insufficient to generate the latent map of x_{dri} because warping can only use existing appearance information in z. When occlusions exist, which are common in those videos containing large motions, LFAE should be able to generate those invisible parts in z. So similar to [62, 76], flow predictor F also estimates a latent occlusion map m with the size of $H_z \times W_z \times 1$, as shown in Fig. 3. Here m contains values changing from 0 to 1 to indicate the degree of occlusion, where 1 is not occluded and 0 means entirely occluded. The final warped latent map $\tilde{z}$ can be produced by:

$$\tilde{z} = m \odot \mathcal{W}(z, f) \ , \quad (5)$$

where $\mathcal{W}(\cdot, \cdot)$ is backward warped map and $\odot$ indicates element-wise multiplication. Decoder Ω subsequently decodes the visible parts and inpaints the occluded parts of latent $\tilde{z}$ for generating output image $\hat{x}_{\text{out}}$, which should be the same as driving frame x_{dri}. Thus LFAE can be trained with the following reconstruction loss using only *unlabeled* video frames:

$$L_{\text{LFAE}} = \mathcal{L}_{\text{rec}}(\hat{x}_{\text{out}}, x_{\text{dri}}) \ , \quad (6)$$

where $\mathcal{L}_{\text{rec}}$ is the loss measuring the difference between reconstructed frame $\hat{x}_{\text{out}}$ and ground truth frame x_{dri}. Here we implement $\mathcal{L}_{\text{rec}}$ using the perceptual loss [33] based on pretrained VGG network [64].

3.2.2 Stage Two: Diffusion Model

In stage two, a 3D-UNet-based diffusion model (DM) is trained to synthesize a temporally-coherent latent flow sequence. The trained image encoder Φ and flow predictor F from stage one are required in the training of this stage two. Given an input video $\mathbf{x}_0^K = \{x_0, x_1, \ldots, x_K\}$ and its corresponding class condition y, we first compute the latent flow

4

sequence from frame 1 to frame k, $\mathbf{f}_1^K = \{f_1, \ldots, f_K\}$, and the occlusion map sequence $\mathbf{m}_1^K = \{m_1, \ldots, m_K\}$ by applying trained F to estimate f_i and m_i between starting frame x_0 and each frame x_i, where $i = 1, \ldots, K$. The size of $\mathbf{f}_1^K$ and $\mathbf{m}_1^K$ are $K \times H_z \times W_z \times 2$ and $K \times H_z \times W_z \times 1$, respectively. We concatenate $\mathbf{f}_1^K$ and $\mathbf{m}_1^K$ along the channel dimension, which produces a $K \times H_z \times W_z \times 3$ volume $\mathbf{s}_0 = \text{cat}[\mathbf{f}_1^K, \mathbf{m}_1^K]$. Then s_0 is mapped to a standard Gaussian noise volume $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ by gradually adding 3D Gaussian noise through the DDPM forward process. The encoder Φ further represents starting frame x_0 as latent map z_0 and pretrained BERT [14] encodes class condition y as text embedding e. Here we choose BERT embedding instead of one-hot encoding to represent y because text embedding can better capture the semantic similarity among different classes. Conditioned on z_0 and e, denoising model $\epsilon_\theta(\mathbf{s}_t, t, z_0, e)$ is trained to predict the added noise ϵ in $\mathbf{s}_t$ based on a conditional 3D U-Net with the following loss:

$$L_{\text{DM}} = \mathbb{E}_{t \sim \mathcal{U}(1,T), \mathbf{s}_0 \sim q(\mathbf{s}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}[||\epsilon - \epsilon_\theta(\mathbf{s}_t, t, z_0, e)||^2] \ , \tag{7}$$

where time step t is uniformly sampled from $\{1, \ldots, T\}$. ϵ_θ is further used in DDPM reverse sampling process to output $\hat{\mathbf{s}}_0 = \text{cat}[\hat{\mathbf{f}}_1^K, \hat{\mathbf{m}}_1^K]$ with the size of $K \times H_z \times W_z \times 3$, where $\hat{\mathbf{f}}_1^K = \{\hat{f}_1, \ldots, \hat{f}_K\}$ and $\hat{\mathbf{m}}_1^K = \{\hat{m}_1, \ldots, \hat{m}_K\}$ are synthesized latent flow and occlusion map sequences. So the *latent flow space* in our DM is only of the size $K \times H_z \times W_z \times 3$. Its dimensions can be much lower than the RGB pixel space dimensions ($K \times H_x \times W_x \times 3$) used by DMs in [24, 27], if the spatial size of latent map $H_z \times W_z$ is much smaller than image size $H_x \times W_x$. Also, the latent flow space only contains motion and shape features. So it can be easier to model than the latent feature space used in [57], which contains other spatial details such as texture and color, as well as all the information in our latent flow space. Therefore, the latent flow space in our approach is both *simple* and *low-dimensional*, which helps to ease the generation and reduce computational cost, as shown in Sec. 4.3.

3.3. Inference

As Fig. 2 shows, for a given image x_0 and condition y, the image encoder Φ encodes x_0 as latent map z_0 and pretrained BERT represents y as embedding e. Conditioned on z_0 and e, a randomly sampled Gaussian noise volume $\mathbf{n}$ with the size of $K_z \times H_z \times W_z \times 3$ is gradually denoised by ϵ_θ through the DDPM reverse sampling process to generate the latent flow sequence $\hat{\mathbf{f}}_1^K$ and occlusion map sequence $\hat{\mathbf{m}}_1^K$. Then the latent map z_0 is warped by each $\hat{f}$ in $\hat{\mathbf{f}}_1^K$ and each $\hat{m}$ in $\hat{\mathbf{m}}_1^K$ according to Eq. 5, producing a new latent map sequence $\tilde{z}_1^K = \{\tilde{z}_1, \ldots, \tilde{z}_K\}$. Finally, each $\tilde{z}$ in $\tilde{z}_1^K$ is further fed to the image decoder Ω for synthesizing each new frame $\hat{x}$ in output video $\hat{\mathbf{x}}_1^K$. Note that flow predictor F is not required during inference.

4. Experiments

4.1. Datasets and Metrics

We conduct comprehensive experiments on the following video datasets.

MUG facial expression dataset [1] contains 1,009 videos of 52 subjects performing 7 different expressions, including *anger*, *disgust*, *fear*, *happiness*, *neutral*, *sadness*, and *surprise*. We randomly select 26 subjects for training and use the remaining 26 subjects for testing, which results in 465 and 544 videos in training and testing set, respectively.

MHAD human action dataset [11] contains 861 videos of 27 actions performed by 8 subjects. This dataset consists of multiple kinds of human actions covering sport actions (*e.g.*, *bowling*), hand gestures (*e.g.*, *draw x*), daily activity (*e.g.*, *stand to sit*) and training exercises (*e.g.*, *lunge*). We randomly choose 4 subjects for training (431 videos) and 4 subjects for testing (430 videos).

NATOPS aircraft handling signal dataset [69] includes 9,600 videos of 20 subjects performing 24 body-and-hand gestures used for communicating with the U.S. Navy pilots. It contains some common handling signals such as *spread wings* and *stop*. We randomly select 10 subjects for training (4,800 videos) and 10 subjects for testing (4,800 videos).

Data Preprocessing. We resize all the videos to 128×128 resolution for our models. For MHAD and NATOPS datasets, we also crop video frames by removing some part of background using their provided depth maps. Considering most videos in these datasets are short (no more than 80 frames on average), instead of uniformly sampling frames, for each training video, we randomly sample 40 frames and sort them by time to generate diverse video clips with fixed length for training the stage-two DM.

Metrics. Following prior works [24, 27, 30, 66], we compute Fréchet Video Distance (**FVD**) [71] to evaluate the *visual quality*, *temporal coherence* and *sample diversity* of generated videos. Similar to Fréchet Inception Distance (FID) [23] used for image quality evaluation, FVD first employs a video classification network I3D [9] pretrained on Kinetics-400 dataset [34] to obtain feature representation of real and synthesized videos. Then it calculates the Fréchet distance between the distributions of real and synthesized video features. To measure how well a generated video corresponds to the class condition y (*condition accuracy*) and the given image x_0 (*subject relevance*), similar to the conditional FID in [6], we design two variants of FVD: class conditional FVD (**cFVD**) and subject conditional FVD (**sFVD**). cFVD and sFVD compare the distance between real and synthesized video feature distributions under the same class condition y or the same subject image x_0, respectively. We first compute cFVD and sFVD for each condition y and image x_0, then report their mean and variance as final results. In our experiments, we generate 10,000 videos for all the
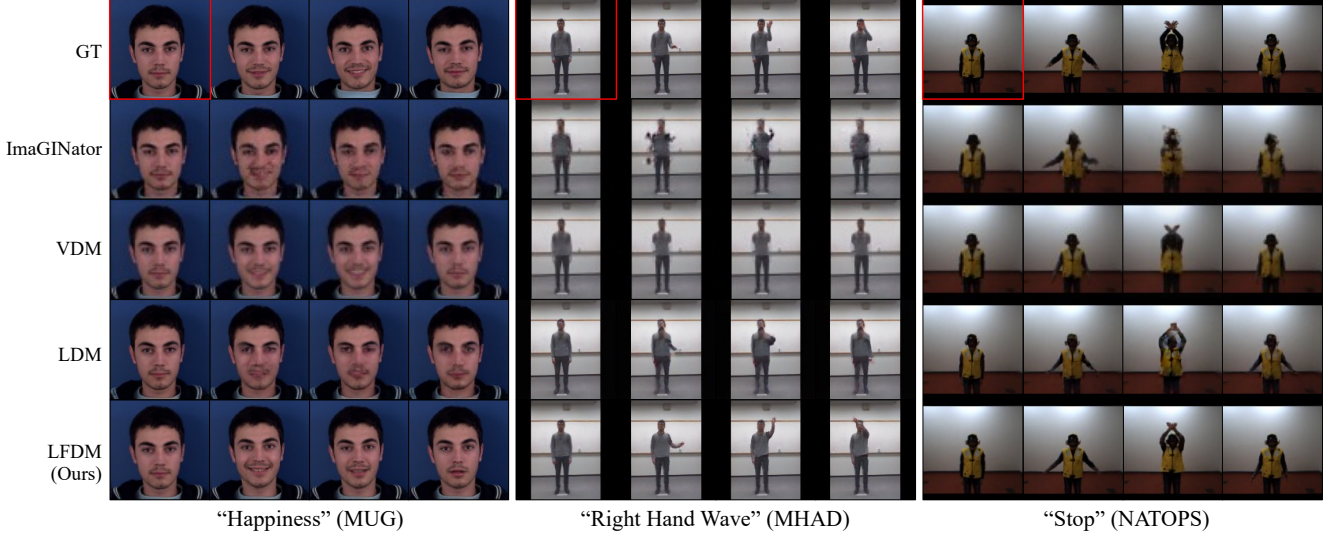
Figure 4. Qualitative comparison among different methods on multiple datasets for cI2V generation. First image frame x_0 is highlighted with red box and condition y is shown under each block. To simplify coding, all the models are designed to also generate starting frame $\hat{x}_0$. The video frames of GT (ground truth), LDM and LFDM have 128×128 resolution while results of ImaGINator and VDM are 64×64.

Model	MUG			MHAD			NATOPS		
	FVD↓	cFVD↓	sFVD↓	FVD↓	cFVD↓	sFVD↓	FVD↓	cFVD↓	sFVD↓
ImaGINator [77]	170.73	257.46±62.88	319.37±95.23	889.48	1406.56±260.70	1175.74±327.99	721.17	1122.13±150.74	1042.69±416.16
VDM [27]	108.02	182.90±69.56	213.59±97.70	295.55	531.20±104.25	398.09±121.16	169.61	410.71±105.97	350.59±125.03
LDM$_{64}$ [57]	123.88	196.49±66.99	236.26±76.08	280.26	515.29±125.70	427.03±112.31	251.72	506.40±125.08	491.37±231.85
LFDM$_{64}$ (Ours)	**27.57**	**77.86±20.27**	**108.36±39.60**	**152.48**	**339.63±52.88**	**242.61±28.50**	**160.84**	**376.14±106.13**	**324.45±116.21**
LDM$_{128}$ [57]	126.28	208.03±64.86	241.49±75.18	337.43	594.34±150.31	497.50±110.16	344.81	627.84±169.52	623.13±320.85
LFDM$_{128}$ (Ours)	**32.09**	**84.52±24.81**	**114.33±42.62**	**214.39**	**426.10±63.48**	**328.76±34.42**	**195.17**	**423.42±117.06**	**369.93±159.26**

Table 1. Quantitative comparison among different methods on multiple datasets for cI2V generation. The 64 and 128 in the subscript of LDM and LFDM indicate that the resolution of synthesized video frames are 64×64 and 128×128, respectively.

models to accurately estimate the feature distributions.

4.2. Model and Baseline Implementation

Model Implementation. Our proposed LFDM includes four trainable modules: an image encoder Φ, an image decoder Ω, a flow predictor F, and a denoising model ϵ_θ from DDPM. These modules are general and can have various backbone networks. Here we choose the architecture in [33] to implement the encoder Φ and the decoder Ω due to its simplicity. For encoder Φ, we use a network with 2 down-sampling blocks, thus the spatial size of latent map $H_z \times W_z$ will be $\frac{1}{4}H_x \times \frac{1}{4}W_x$, only 1/16 of the size of the input frame x. For decoder Ω, we use a network with 6 residual blocks and two up-sampling blocks. The flow predictor F is implemented with MRAA [63], which can estimate latent flow f and occlusion map m based on detected object parts. Per MRAA [63], we also add the equivariance loss to L_{LFAE} in Eq. 6. When training LFAE, we set the batch size to be 100 and train it for 100 epochs using the Adam optimizer [38]. The initial learning rate is set to be 2×10^{-4} and drops by a decay factor 0.1 at epoch 60 and 90.

For the denoising model ϵ_θ in stage-two DM, we adopt

the conditional 3D U-Net architecture in [27], which includes 4 down-sampling and 4 up-sampling 3D covolutional blocks. The embedding e of the condition y is concatenated with a time step embedding and then added into each residual blocks of ϵ_θ. The latent map z_0 of image x_0 is also provided to ϵ_θ by the concatenation with the noise $\mathbf{n}$. When training DM, we set the batch size to be 20 and train 1,200 epochs using the Adam optimizer [38]. The initial learning rate is set to be 2×10^{-4} and drops by a decay factor 0.1 at epoch 800 and 1000. We employ the cosine noise schedule [52] and use dynamic thresholding [61] with 90 percentile during the sampling process. To enable stochastic generation, we adopt the training process similar to classifier-free guidance [26], *i.e.*, the condition embedding e is replaced with a null embedding $\emptyset$ for ϵ_θ with 10% probability. So stochastic generation can be achieved by simply feeding $\emptyset$ instead of e to ϵ_θ during inference (*i.e.*, $g = 0$ in Eq. 4). But we still set $g = 1$ for conditional generation instead of using the common $g > 1$, because the latter will run two DM forward passes, leading to slower sampling speed [26]. To simplify coding, the flow sequence $\hat{\mathbf{f}}$ and occlusion map sequence $\hat{\mathbf{m}}$ also include f_0 and m_0 (*i.e.*, the flow and oc-

Model	FVD↓		Gap↓
	Train	Test	
ImaGINator [77]	15.92	170.73	154.81
VDM [27]	17.39	108.02	90.63
LDM_{64} [57]	55.44	123.88	68.44
$LFDM_{64}$ (Ours)	**10.58**	**27.57**	**16.99**
LDM_{128} [57]	67.51	126.28	58.77
$LFDM_{128}$ (Ours)	**14.17**	**32.09**	**17.92**

Table 2. Comparison of different models conditioned on image x_0 from either the training set or testing set of MUG dataset.

clusion map between x_0 and itself). So $\hat{\mathbf{f}}$ and $\hat{\mathbf{m}}$ can have the same size as the output video $\hat{\mathbf{x}}_0^K$. Unless otherwise specified, we apply $T = 1000$ step DDPM to sample 40-frame $32 \times 32 \times 2$ $\hat{\mathbf{f}}$ and $32 \times 32 \times 1$ $\hat{\mathbf{m}}$ and finally produce 40-frame videos $\hat{\mathbf{x}}$ with 128×128 frame resolution.

Baseline Implementation. We compare LFDM with three baseline models, including GAN-based I2V model **ImaGINator** [77], video diffusion models **VDM** [27], and a variant of image latent diffusion models **LDM** [78]. We extend LDM to the video domain by replacing the 2D U-Net in their DM with 3D U-Net. For a fair comparison, VDM, LDM and our LFDM use the same 3D U-Net architecture and the same way to utilize conditioning information by concatenation. LDM is designed to have the same-size latent space as LFDM ($40 \times 32 \times 32 \times 3$) and also generate 40-frame videos with 128×128 resolution. Due to high computational cost, we only use VDM to generate 40-frame videos with 64×64 resolution. ImaGINator is implemented with their provided code, which generates 32-frame videos with 64×64 resolution. For a fair comparison, when computing metrics, all the real videos are resized to be the same volume size as the generated videos for each model. We also report the results of LDM and LFDM under the 64×64 resolution. For sampling, we use 1000-step DDPM for LDM and LFDM. Since DDPM sampling is very slow in the large latent space of VDM ($40 \times 64 \times 64 \times 3$), we employ 200-step DDIM [68] to accelerate the sampling process. We find that this method can achieve comparable performance as DDPM for VDM in our preliminary experiments.

4.3. Result Analysis

Conditional Generation. Table 1 shows the quantitative comparison between our method and baseline models for conditional image-to-video (cI2V) task, where our proposed LFDM consistently outperforms all the other baseline models under the resolution of either 64×64 or 128×128. As Fig. 4 shows, ImaGINator and LDM both fail to produce fine-grained details while VDM sometimes just generates some almost-still frame sequences (*e.g.*, results on MHAD in Fig 4). ImaGINator may suffer from the limited synthesis ability of their used GAN model, while LDM and VDM may sometimes have difficulty in modeling their latent spaces, which couple spatial and temporal features.

Model	L_1 error↓		
	MUG	MHAD	NATOPS
LDM [57]	0.419	**0.294**	**0.222**
LFDM (Ours)	**0.418**	0.302	0.260

Table 3. Comparison of self-reconstruction L_1 error between the auto-encoder of LDM and that of LFDM on testing sets of multiple datasets under the resolution of 128×128.

Model	FVD↓	sFVD↓
ImaGINator [77]	196.18	352.52±104.54
VDM [27]	106.66	214.21±95.96
LDM_{64} [57]	125.36	242.19±79.40
$LFDM_{64}$ (Ours)	**42.50**	**154.50±50.54**
LDM_{128} [57]	127.12	247.28±74.26
$LFDM_{128}$ (Ours)	**49.31**	**167.58±56.85**

Table 4. Comparison among different methods on MUG dataset for stochastic image-to-video generation.

Though VDM achieves similar FVD scores to ours on NATOPS dataset, note that VDM has higher computational cost than LFDM since the latent flow space in our DM has lower dimensions. When sampling with the batch size of 10 using 1000-step DDPM on one NVIDIA A100 GPU, LFDM costs about 0.9GB and 36s to generate one video of 128×128 resolution while VDM requires about 2.5GB and 112.5s to sample one video of only 64×64 resolution.

To further analyze the performance difference among different models, we also compute their FVD scores of generated videos conditioned on the image x_0 from the training set of MUG dataset. As Table 2 shows, all three baseline models have much better performance when conditioned on training (*seen*) images than testing (*unseen*) images, while our proposed LFDM noticeably shows the least training-testing gap. This can be attributed to our warp-based design, two-stage disentangled training framework, among others. We also notice that LDM fails to achieve low FVD scores like other methods when conditioned on training images. Considering that the auto-encoder in LDM has already achieved similar or even better reconstruction performance than LFDM (see Table 3), we speculate that the 3D U-Net in LDM may not be effective enough to model the latent space that encodes both spatial and temporal features. However, built on the same 3D U-Net architecture, LFDM can successfully synthesize visually-appealing and temporally-coherent videos, illustrating simpler latent space in our DM, which only contains motion and shape information. From Table 1, one can also observe that all the models achieve better FVD results on MUG dataset than on MHAD and NATOPS datasets. The reason may be that facial videos contain less high-frequency spatio-temporal details than human movement videos.

Stochastic Generation. As mentioned in Sec. 4.2, by replacing the condition embedding e with the null embedding $\emptyset$ during inference, we enable VDM, LDM, and our

Figure 5. Qualitative comparison of LFDM with original (the 1st&3rd rows) *vs.* finetuned (the 2nd&4th rows) decoder on Face-Forensics dataset [59]. The first column shows the given image x_0 and condition y. The green boxes highlight differences.

Model	L_1 error↓	FID↓	sFID↓
Original decoder Ω	2.129	38.48	75.08±37.33
Finetuned decoder Ω	**1.310**	**23.36**	**51.56±26.36**

Table 5. Quantitative comparison of LFDM with original *vs.* fine-tuned decoder on FaceForensics dataset.

Steps	g-scale	FVD↓	Time(s)↓
DDIM-10	1.0	50.18	**0.3**
DDIM-100	1.0	106.11	3.5
DDPM-1000	1.1	**31.84**	71.4
DDPM-1000	5.0	49.82	71.4
DDPM-1000	1.0	32.09	36.0

Table 6. Ablation study comparing different sampling strategies for LFDM on MUG dataset (resolution is 128×128).

LFDM for stochastic generation that only depends on image x_0. We also retrain ImaGINator for stochastic generation by removing the condition input y. As Table 4 shows, our proposed LFDM still outperforms all the other baseline models on MUG dataset, validating the effectiveness of LFDM on both conditional and stochastic I2V generation.

Application to New-domain Facial Videos. We also apply our proposed LFDM trained on MUG dataset to new-domain facial videos. Here we choose **FaceForensics** [59] dataset, which contains 1,004 subject videos of news briefing from different reporters. We randomly choose 150 subjects for training and 150 subjects for testing, and we utilize a face alignment algorithm [37] to extract the facial regions of video frames. We compare two models: (1) directly applying the original trained LFDM, and (2) finetuning the decoder Ω with the unsupervised training loss L_{LFAE} in Eq. 6 using the training videos from FaceForensics dataset, but freezing all the remaining modules. Due to the lack of ground truth videos, to measure performance, we instead calculate image-based **FID** [23] and subject FID (**sFID**) scores. Similar to sFVD, sFID first computes the FID scores between distributions of real and synthesized frames for each subject image x_0 and then reports their mean and variance. We also report the L_1 error on testing set to compare the self-reconstruction performance of LFAE using original and finetuned Ω. As Table 5 and Fig. 5 show, by simply finetuning the decoder Ω with *unlabeled* new videos, LFDM can still achieve promising performance on new-domain facial videos. This illustrates the flexibility of our two-stage training framework. To improve spatial content quality, one can just finetune the decoder in stage-one LFAE, without the need to retrain the whole framework.

Ablation Study. We conduct ablation study of differ-ent sampling strategies in Table 6. For each setting, we report FVD scores and the average sampling time to generate one video when using the batch size of 10 on one NVIDIA A100 GPU. We first compare the results sampled by 10-step DDIM and 100-step DDIM against 1000-step DDPM (default setting). It is interesting to observe that 10-step DDIM shows noticeably better performance than 100-step DDIM with faster sampling speed. We also try two typical values of guidance scale g in Eq. 4, small guidance 1.1 and large guidance 5.0. Though using $g = 1.1$ guidance slightly improves the FVD score, it also doubles the inference time. So we use $g = 1$ as our default setting. We also try to change the network depth of the decoder Ω in stage-one LFAE and the denoising model ϵ_θ in stage-two DM; results of this ablation study, generated video examples and further discussion are included in our Supp. materials.

5. Conclusion and Discussion

In this paper, we propose a novel conditional image-to-video framework, latent flow diffusion models (LFDM), to generate videos by warping given images with flow sequences generated in the latent space based on class conditions. Comprehensive experiments show that LFDM can achieve state-of-the-art performance on multiple datasets.

Though achieving promising performance, our proposed LFDM still suffers from several limitations. First, current experiments with LFDM are limited to videos containing a single moving subject. We plan to extend the application of LFDM to multi-subject flow generation in the future. Second, the current LFDM is conditioned on class labels instead of natural text descriptions. Text-to-flow is an interesting topic and we leave this direction as future work. Finally, compared with GAN models, LFDM is much slower when sampling with 1000-step DDPM. In the future, we plan to further investigate some fast sampling methods [40, 45] in order to reduce generation time.

References

[1] Niki Aifanti, Christos Papachristou, and Anastasios Delopoulos. The mug facial expression database. In *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10*, pages 1–4. IEEE, 2010. 5

[2] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. *arXiv preprint arXiv:2111.14818*, 2021. 3

[3] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine. Stochastic variational video prediction. *arXiv preprint arXiv:1710.11252*, 2017. 2

[4] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International journal of computer vision*, 92(1):1–31, 2011. 1

[5] Yogesh Balaji, Martin Renqiang Min, Bing Bai, Rama Chellappa, and Hans Peter Graf. Conditional gan with discriminative filter generation for text-to-video synthesis. In *IJCAI*, volume 1, page 2, 2019. 2

[6] Yaniv Benny, Tomer Galanti, Sagie Benaim, and Lior Wolf. Evaluation metrics for conditional image generation. *International Journal of Computer Vision*, 129(5):1712–1731, 2021. 5

[7] Andreas Blattmann, Timo Milbich, Michael Dorkenwald, and Björn Ommer. ipoke: Poking a still image for controlled stochastic video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14707–14717, 2021. 2, 3

[8] Andreas Blattmann, Timo Milbich, Michael Dorkenwald, and Bjorn Ommer. Understanding object dynamics for interactive image-to-video synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5171–5181, 2021. 2, 3

[9] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 5

[10] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5933–5942, 2019. 2

[11] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. 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)*, pages 168–172. IEEE, 2015. 5

[12] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432. Springer, 2016. 2, 3

[13] Kangle Deng, Tianyi Fei, Xin Huang, and Yuxin Peng. Ircgan: Introspective recurrent convolutional gan for text-to-video generation. In *IJCAI*, pages 2216–2222, 2019. 2

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 5

[15] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[16] Michael Dorkenwald, Timo Milbich, Andreas Blattmann, Robin Rombach, Konstantinos G Derpanis, and Bjorn Ommer. Stochastic image-to-video synthesis using cinns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3742–3753, 2021. 1, 2

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3

[18] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. *arXiv preprint arXiv:2111.14822*, 2021. 3

[19] Ligong Han, Jian Ren, Hsin-Ying Lee, Francesco Barbieri, Kyle Olszewski, Shervin Minaee, Dimitris Metaxas, and Sergey Tulyakov. Show me what and tell me how: Video synthesis via multimodal conditioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3615–3625, 2022. 1

[20] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *arXiv preprint arXiv:2205.11495*, 2022. 2, 3

[21] Jiawei He, Andreas Lehrmann, Joseph Marino, Greg Mori, and Leonid Sigal. Probabilistic video generation using holistic attribute control. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–467, 2018. 1

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5, 8

[24] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2, 3, 5

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 3

[26] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 3, 6

[27] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 2, 3, 5, 6, 7

[28] Aleksander Holynski, Brian L Curless, Steven M Seitz, and Richard Szeliski. Animating pictures with eulerian motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5810–5819, 2021. 3

[29] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[30] Yaosi Hu, Chong Luo, and Zhenzhong Chen. Make it move: controllable image-to-video generation with text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18219–18228, 2022. 1, 2, 3, 5

[31] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015. 4

[32] Yuming Jiang, Shuai Yang, Haonan Qiu, Wayne Wu, Chen Change Loy, and Ziwei Liu. Text2human: Text-driven controllable human image generation. *arXiv preprint arXiv:2205.15996*, 2022. 3

[33] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 4, 6

[34] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 5

[35] Gwanghyun Kim and Jong Chul Ye. Diffusionclip: Text-guided image manipulation using diffusion models. 2021. 3

[36] Yunji Kim, Seonghyeon Nam, In Cho, and Seon Joo Kim. Unsupervised keypoint learning for guiding class-conditional video prediction. *Advances in neural information processing systems*, 32, 2019. 1, 2, 3

[37] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009. 8

[38] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[39] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[40] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. *arXiv preprint arXiv:2106.00132*, 2021. 8

[41] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Flow-grounded spatial-temporal video prediction from still images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 600–615, 2018. 2, 3

[42] Yitong Li, Martin Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video generation from text. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018. 2

[43] Zhiheng Li, Martin Renqiang Min, Kai Li, and Chenliang Xu. Stylet2i: Toward compositional and high-fidelity text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18197–18207, 2022. 1

[44] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. 2022. 3

[45] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022. 8

[46] Aniruddha Mahapatra and Kuldeep Kulkarni. Controllable animation of fluid elements in still images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3667–3676, 2022. 2, 3

[47] Arun Mallya, Ting-Chun Wang, Karan Sapra, and Ming-Yu Liu. World-consistent video-to-video synthesis. In *European Conference on Computer Vision*, pages 359–378. Springer, 2020. 2

[48] Tanya Marwah, Gaurav Mittal, and Vineeth N Balasubramanian. Attentive semantic video generation using captions. In *Proceedings of the IEEE international conference on computer vision*, pages 1426–1434, 2017. 2

[49] Gaurav Mittal, Tanya Marwah, and Vineeth N Balasubramanian. Sync-draw: Automatic video generation using deep recurrent attentive architectures. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1096–1104, 2017. 2

[50] Haomiao Ni, Yihao Liu, Sharon X. Huang, and Yuan Xue. Cross-identity video motion retargeting with joint transformation and synthesis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 412–422, January 2023. 2

[51] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[52] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 3, 6

[53] Junting Pan, Chengyu Wang, Xu Jia, Jing Shao, Lu Sheng, Junjie Yan, and Xiaogang Wang. Video generation from single semantic label map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2019. 2, 3

[54] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1789–1798, 2017. 2

[55] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10619–10629, 2022. 3

[56] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 3

[57] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2, 3, 5, 6, 7

[58] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

[59] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179*, 2018. 8

[60] Chitwan Saharia, William Chan, Huiwen Chang, Chris A Lee, Jonathan Ho, Tim Salimans, David J Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. *arXiv preprint arXiv:2111.05826*, 2021. 3

[61] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 3, 6

[62] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in Neural Information Processing Systems*, 32, 2019. 2, 4

[63] Aliaksandr Siarohin, Oliver J Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13653–13662, 2021. 6

[64] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[65] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2, 3

[66] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3626–3636, 2022. 5

[67] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[68] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 7

[69] Yale Song, David Demirdjian, and Randall Davis. Tracking body and hands for gesture recognition: Natops aircraft handling signals database. In *2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, pages 500–506. IEEE, 2011. 5

[70] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[71] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 5

[72] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[73] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Masked conditional video diffusion for prediction, generation, and interpolation. *arXiv preprint arXiv:2205.09853*, 2022. 2

[74] Vikram Voleti, Alexia Jolicoeur-Martineau, and Christopher Pal. Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation. *CoRR*, 2022. 3

[75] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. *arXiv preprint arXiv:1910.12713*, 2019. 2

[76] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*, 2018. 2, 4

[77] Yaohui Wang, Piotr Bilinski, Francois Bremond, and Antitza Dantcheva. Imaginator: Conditional spatio-temporal gan for video generation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1160–1169, 2020. 1, 2, 3, 6, 7

[78] Yaohui Wang, Di Yang, Francois Bremond, and Antitza Dantcheva. Latent image animator: Learning to animate images via latent space navigation. *arXiv preprint arXiv:2203.09043*, 2022. 2, 7

[79] Olivia Wiles, A Koepke, and Andrew Zisserman. X2face: A network for controlling face generation using images, audio, and pose codes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–686, 2018. 2

[80] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021. 2

[81] Wei Xiong, Wenhan Luo, Lin Ma, Wei Liu, and Jiebo Luo. Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2364–2373, 2018. 2

[82] Yuan Xue, Yuan-Chen Guo, Han Zhang, Tao Xu, Song-Hai Zhang, and Xiaolei Huang. Deep image synthesis from intuitive user input: A review and perspectives. *Computational Visual Media*, 8:3–31, 2022. 1

[83] Ceyuan Yang, Zhe Wang, Xinge Zhu, Chen Huang, Jianping Shi, and Dahua Lin. Pose guided human video generation.

In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018. 2, 3

[84] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*, 2022. 2, 3

[85] Jiangning Zhang, Chao Xu, Liang Liu, Mengmeng Wang, Xia Wu, Yong Liu, and Yunliang Jiang. Dtvnet: Dynamic time-lapse video generation via single still image. In *European Conference on Computer Vision*, pages 300–315. Springer, 2020. 2, 3

A1. Potential Negative Social Impact

Conditional image-to-video models can be used for unethical purposes [8], *e.g.*, creating videos of celebrities for fake news spreading. We will restrict the usage of our models to research purposes only. We also plan to investigate some fake video detection techniques [1] that may be effective in detecting fake videos like the ones generated by our methods.

A2. Additional Experiments

A2.1. Additional Ablation Study on Network Architecture

To evaluate the performance difference of our proposed LFDM with different architectures, we change the depth of the image decoder Ω in stage-one LFAE (Table A1) and the 3D U-Net ϵ_θ in stage-two DM (Table A2). We experiment with different settings on MUG dataset to generate videos of 128×128 frame resolution.

In our default setting, the image decoder Ω in stage-one LFAE is implemented with a network including 6 residual blocks and 2 up-sampling blocks. In Table A1, we compare using different network depths for the image decoder Ω in stage-one LFAE . We add four extra residual blocks to the decoder Ω. So the number of residual blocks is increased from 6 to 10. Then we only retrain this deeper decoder in stage one, while keeping all the remaining modules unchanged. As Table A1 shows, using a deeper image decoder shows slightly better self-reconstruction performance (as measured by L_1 error) but fails to generate higher-quality videos (as measured by FVD). Therefore, we keep using 6 residual blocks in our experiments.

In our default setting, the denoising network ϵ_θ employs a 3D U-Net architecture including 4 down-sampling and 4 up-sampling 3D convolutional blocks, where the *channel multipliers* are (1, 2, 4, 8) with a base channel of 64. That is, from highest to lowest resolution, the 4 down- or up-sampling blocks in ϵ_θ use (1×64, 2×64, 4×64, 8×64) channels, respectively. In Table A2, we compare using different channel multipliers in stage-two DM . We add one more layer to the down-sampling and up-sampling blocks of the 3D U-Net and the channel multipliers are (1, 2, 4, 8,

# Residual Blocks	L_1 error↓	FVD↓
6	0.418	**32.09**
10	**0.371**	32.83

Table A1. Comparison using different numbers of residual blocks in the image decoder Ω of stage-one LFAE.

Channel Multipliers	FVD↓
(1, 2, 4, 8)	**32.09**
(1, 2, 4, 8, 16)	68.07

Table A2. Comparison using different channel multipliers in the network ϵ_θ of stage-two DM.

16) with a base channel of 64. We retrain this deeper DM in stage two with 1,200 training epochs as in our previous simpler DM training. We keep using the same stage-one LFAE. From Table A2, one can observe that using more layers in DM led to decreased performance. Therefore, we adopt the simpler (1, 2, 4, 8) as the default setting of channel multipliers in our stage-two DM.

A2.2. Additional Analysis of Flow and Occlusion Maps

Figure A1. Visualization of flow and occlusion map. [Ref. + Flow] is generated by applying the flow to reference image; note the change in head pose and the shape of eyes and mouth after applying flow. Occlusion map further masks the eyes and mouth to help decoder generate novel pixels for these parts in output image.

Figure A1 shows the visualization of flow and occlusion map of one example video frame from MUG dataset. As illustrated in the caption of Fig. A1, without using occlusion map, decoder may need to learn which regions should be kept and which regions should be masked and repainted. Our additional experiments show that retraining LFAE without occlusion maps increases the L_1 error of self-reconstruction from 0.418 to 0.450 on MUG dataset.

A2.3. Comparison of Inference Time among Different Models

Table A3 shows the average inference time of each method to generate one video when using batch size 10 on one NVIDIA A100 GPU on MUG dataset. Note that VDM uses 200-step DDIM while both LDM and LFDM employ 1000-step DDPM.

Model	ImaGINator	VDM	LDM$_{64}$	LFDM$_{64}$	LDM$_{128}$	LFDM$_{128}$
Time(s)	0.9	23.1	8.0	8.8	25.5	36.0

Table A3. Inference time comparison among different methods.

A3. More Discussion about Future Work

Several limitations and some future work are discussed in Section 5 of the paper. Here we elaborate more on future work about LFDM. One future direction is to enable the generation of a video with changing background (or context). We plan to first utilize our LFDM to generate a video describing the motion of foreground subject, and then design another generative network conditioned on each generated foreground frame to synthesize the changing background for each frame. In addition, to enhance the generalization ability of LFDM on generating diverse motions of more categories, we plan to collect more labeled training video datasets and apply some continual/incremental learning techniques such as [4–7] to train our LFDM. Finally, in our experiments (Table 6), we noticed that 10-step DDIM can achieve acceptable generation performance with faster sampling speed, suggesting it may have greater potential with better hyperparameter settings. To explore these settings, including diffusion sampling steps, we plan to employ some recent hyperparameter optimization techniques such as [2, 3, 9].

A4. Information about Attached Videos

We attach seven MP4 files of example video clips generated by our proposed method in Supp. materials[1]. All the given images are testing (*unseen*) images.

- **mug.mp4** shows the synthesized video clips displaying all 7 expressions of one subject from MUG dataset.

- **mhad1.mp4** and **mhad2.mp4** include the generated video clips for 26 actions of one subject from MHAD dataset. We exclude the action *sit to stand* because the subject in the given image is standing.

- **natops.mp4** shows the synthesized video clips containing all 24 gestures of one subject from NATOPS dataset.

---

[1]These videos are also available in https://github.com/nihaomiao/CVPR23_LFDM.

- **new_domain.mp4** shows the synthesized video clips including 4 expressions of four subjects from FaceForensics dataset. "Original" means directly applying our LFDM pretrained on MUG dataset. "Finetuned" means that the image decoder is finetuned with the *training* videos from FaceForensics dataset. Note that other modules including stage-two DM are still unchanged during finetuning. From this video, one can observe that our original LFDM can generate acceptable results for given subject images from a new domain and achieve better performance when the decoder is finetuned with training videos from the new domain.

- **mug_ddim.mp4** shows the synthesized video clips containing 4 expressions of four subjects from MUG dataset. "DDIM-10" means using 10-step DDIM for diffusion sampling while "DDPM-1000" is our default 1000-step DDPM sampling strategy. From this video, one can observe that 10-step DDIM can generate visually-acceptable videos with faster sampling speed (0.3s per video *vs*. 36s per video when using DDPM-1000). But note that the FVD score of DDPM-1000 is still noticeably better than DDIM-10 (32.09 *vs*. 50.18) so we keep DDPM-1000 as our default setting.

- **sota.mp4** is a video for comparison between our proposed LFDM and several other models including ImaGINator, VDM, and LDM. We show synthesized video clips by each model on 3 subjects from MUG, MHAD, and NATOPS datasets. The video frames of ground truth (GT) and results of LDM and our LFDM have 128×128 resolution while results of ImaGINator and VDM are 64×64. The original video clips generated by ImaGINator only contain 32 frames. So we repeat the first frame and the last frame four times to make all the displaying videos have 40 frames.

References

[1] Irene Amerini, Leonardo Galteri, Roberto Caldelli, and Alberto Del Bimbo. Deepfake video detection through optical flow based cnn. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0, 2019. 1

[2] James Bergstra and Yoshua Bengio. Random search for hyperparameter optimization. *Journal of machine learning research*, 13(2), 2012. 2

[3] Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816, 2017. 2

[4] Mingfu Liang, Jiahuan Zhou, Wei Wei, and Ying Wu. Balancing between forgetting and acquisition in incremental subpopulation learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 364–380. Springer, 2022. 2

[5] Riccardo Volpi, Diane Larlus, and Grégory Rogez. Continual adaptation of visual representations via domain randomization and meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4443–4453, 2021. 2

[6] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 631–648. Springer, 2022. 2

[7] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 2

[8] Sihyun Yu, Jihoon Tack, Sangwoo Mo, Hyunsu Kim, Junho Kim, Jung-Woo Ha, and Jinwoo Shin. Generating videos with dynamics-aware implicit generative adversarial networks. *arXiv preprint arXiv:2202.10571*, 2022. 1

[9] Shaokun Zhang, Feiran Jia, Chi Wang, and Qingyun Wu. Targeted hyperparameter optimization with lexicographic preferences over multiple objectives. In *The Eleventh International Conference on Learning Representations*, 2023. 2