9-2024

Text-driven video prediction

Xue SONG
Fudan University

Jingjing CHEN
Fudan University

Bin ZHU
Singapore Management University, binzhu@smu.edu.sg

Yu-gang JIANG
Fudan University

Citation

SONG, Xue; CHEN, Jingjing; ZHU, Bin; and JIANG, Yu-gang. Text-driven video prediction. (2024). *ACM Transactions on Multimedia Computing, Communications and Applications*. 20, (9), 1-15.
Available at: https://ink.library.smu.edu.sg/sis_research/9356

Text-driven Video Prediction

Xue Song, Jingjing Chen, Bin Zhu, and Yu-Gang Jiang

***Abstract*—Current video generation models usually convert signals indicating appearance and motion received from inputs (e.g., image, text) or latent spaces (e.g., noise vectors) into consecutive frames, fulfilling a stochastic generation process for the uncertainty introduced by latent code sampling. However, this generation pattern lacks deterministic constraints for both appearance and motion, leading to uncontrollable and undesirable outcomes. To this end, we propose a new task called Text-driven Video Prediction (TVP). Taking the first frame and text caption as inputs, this task aims to synthesize the following frames. Specifically, appearance and motion components are provided by the image and caption separately. The key to addressing the TVP task depends on fully exploring the underlying motion information in text descriptions, thus facilitating plausible video generation. In fact, this task is intrinsically a cause-and-effect problem, as the text content directly influences the motion changes of frames. To investigate the capability of text in causal inference for progressive motion information, our TVP framework contains a Text Inference Module (TIM), producing step-wise embeddings to regulate motion inference for subsequent frames. In particular, a refinement mechanism incorporating global motion semantics guarantees coherent generation. Extensive experiments are conducted on Something-Something V2 and Single Moving MNIST datasets. Experimental results demonstrate that our model achieves better results over other baselines, verifying the effectiveness of the proposed framework.**

***Index Terms*—Text-driven Video Prediction, Motion Inference, Controllable Video Generation.**

I. Introduction

Video generation [1]–[5] is a popular topic, drawing numerous research attention in recent years. It aims to create video frames according to input signals (e.g., image or text) or latent spaces (e.g., noise vectors). Existing works on video generation could be divided into two categories: unconditional and conditional video generation. Unconditional video generation [4], [6], [7] tends to produce videos entirely from latent codes (e.g., noises), leading to completely uncontrollable and even undesirable results. In contrast, conditional video generation incorporates several certain factors, like image-to-video generation [8]–[11] and text-to-video generation [12]–[14]. Compared to the unconditional one, it provides certain deterministic information from image or text inputs and achieves semi-controlled video generation. Regardless of the different settings, video generation acts as an important role in real-world applications, like video creation and data augmentation. Therefore, studying the problem of video generation has significant importance.

In the literature, since the video content could be disentangled to appearance and motion factors, existing video generation works generally transform signals representing appearance and motion semantics into consecutive frames, thus reducing the generation difficulty. As the representative works of unconditional video generation, MoCoGAN [6] and G^{3}AN [4] separately sampled these two components from appearance and motion spaces with the form of noise variables, which may result in undesirable combination and deficiency of video quality. Regarded as one type of conditional video generation, image-to-video generation only provides the first image as the appearance factor, while the corresponding motion code is sampled from latent spaces (e.g., noise vectors). However, the generated videos are far from visually pleasant and even totally counterfactual [8], especially for poor motion consistency among frames. Different from image-to-video generation, text-to-video generation gives relatively complete motion description from text modality and leaves more room for appearance construction. Nevertheless, the predicted contents are blurry and even far from real videos. Therefore, the above-mentioned video generation tasks lack complete control on both elements for plausible and deterministic video outcomes.

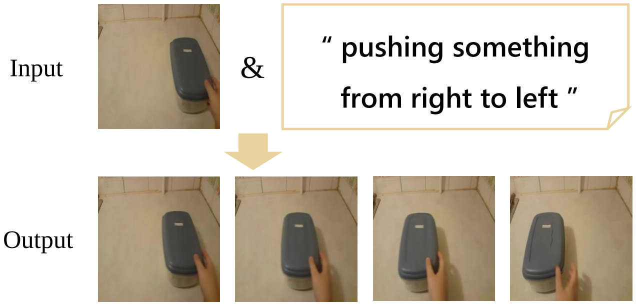

Fig. 1. The definition of Text-driven Video Prediction (TVP) task. Given the image and caption, it aims to generate subsequent frames as the text guided. Specifically, appearance and motion components are provided by the image and caption separately to produce a deterministic and controllable video.

To provide guidance for both appearance and motion semantics to achieve totally controllable generation, we propose the Text-driven Video Prediction (TVP) task. As illustrated in Fig. 1, TVP takes the first frame and caption describing motion changes as inputs, then outputs the subsequent frames. Specifically, the input image offers appearance while motion is supplied by the text description, thus expecting to produce deterministic video frames. From the perspective of real-world applications, given an image, users could offer their intentions for what will happen next as the origin of text descriptions. Then the TVP produces the desirable video and achieves controllable generation, which realizes customization for users and owns promising prospects. The major challenge of this task is how to integrate motion information contained in text into visual images which should also transit harmoniously in

Xue Song, Jingjing Chen and Yu-Gang Jiang are with Fudan University, Shanghai, China (e-mail: 18210860029, chenjingjing, ygj@fudan.edu.cn). Bin Zhu is with City University of Hong Kong, Kowloon, Hong Kong (e-mail: andrewzhu1216@gmail.com).

the temporal dimension. To tackle the above problem, this paper explores the TVP from a new analysis perspective, i.e., the cause-and-effect. More concretely, the position change of the object in the first image totally depends on the instruction in the text description. For example, in Fig. 1, the caption is “moving something from right to left", thus in the following frames, the box in the first frame should be progressively moved towards the left direction. Therefore, this problem could be regarded as how to explore the role of text in causal inference for motion evolution in predicted videos. Finally, combining appearance and text-driven motion, the desirable and controllable video is produced. Overall, the TVP task provides strong constraints to fulfill deterministic video generation, which is more plausible than existing generation tasks and could better satisfy the users’ generation requirements.

For the TVP task, we design the GAN-based framework as shown in Fig. 2. It mainly contains two parts: motion inference and video generation. Motion inference investigates the causality in text and predicts the motion for objects presented in the first frame, while the video generation predicts consecutive video frames with a pre-trained generator using the latent codes obtained from the first frame and predicted motions. More specifically, motion inference contains two steps: text inference and motion prediction. Especially, the Text Inference Module (TIM) in text inference is designed to fully investigate causality in text for motion information. Concretely, considering words in text weigh differently for progressive motion evolution, it first calculates the primary fusion embeddings of word-level representations for each inference step, i.e., each frame. Since such operation is performed independently for steps, it may lead to motion inconsistency across frames. To supply the overall motion information, a refinement mechanism integrates global semantics gained in sentence-level representations to produce the step-wise inference embeddings. Then the motion prediction converts them into motion residuals in a latent space shared by the pre-trained generator. Finally, the latent vectors are used to get predicted frames. To evaluate our framework, we use a synthetic dataset, i.e., Single Moving MNIST [15], and a realistic dataset, i.e. Something-Something V2 [16]. The main contributions of our work are summarized as follows:

- We propose a new task called Text-driven Video Prediction (TVP) aiming at producing controllable and deterministic video frames given the first frame and text description. Especially, appearance and motion components for video generation are provided by the image and caption separately.
- We solve TVP from a causal inference task perspective as the motion change of the objects in frames are directly guided by text descriptions, and propose a GAN-based framework. The essential part is the Text Inference Module (TIM), which generates step-wise inference embeddings for progressive motion prediction.
- We evaluate our framework on two datasets, Single Moving MNIST and Something-Something V2, and it attains better results than baselines, illustrating the effectiveness of our proposed framework.

II. Related Works

This section reviews the existing works on video generation. Specifically, we review two types of video generation tasks, i.e., text-to-video generation and image-to-video generation.

A. Text-to-video Generation

Since the text-to-video generation task inputs text descriptions as guidance to produce semantically consistent videos, it gives no explicit constraint on the corresponding appearance of predicted videos. For text-to-video generation, the first work Sync-DRAW [15] proposed a framework that combined a Variational Autoencoder (VAE) with a Recurrent Attention Mechanism. Specifically, the attention mechanism was responsible for synchronizing frames while the VAE was designed to learn a latent distribution for videos conditioned on text features. By contrast, Li et al. [13] associated VAE with Generative Adversarial Network (GAN). Concretely, a conditional VAE was adopted to form the background color and object layout of the video as gist, facilitating the generator to produce frames. TGANs-C [17] transformed the concatenation of a noise variable and caption embedding into frames via 3D spatio-temporal convolutions. Later, there are works that struggle to strengthen the associations between videos and text descriptions. Apart from conditioning generators with text descriptions, TFGAN [3] also imposed this constraint on discriminators. IRC-GAN [18] further introduced mutual information to quantitatively estimate the semantic similarity between generated videos and texts. Marwah et al. [19] allowed text to combine with short-term video context (i.e., previous frame) and long-term one (i.e., previous frames) through soft-attention. Different from GAN or VAE-based networks, Liu et al. [14] proposed a cross-modal dual learning algorithm, modeling bidirectional transformations between videos and captions. Recently, GODIVA [12] was proposed as a VQ-VAE based pre-trained model, utilizing a three-dimensional sparse attention mechanism to enable operation on spatial and temporal dimensions simultaneously.

B. Image-to-video Generation

Image-to-video generation task aims to produce the whole video from a single image. As the image only offers the appearance factor, the corresponding motion is stochastic. Generally, approaches to handling this task could be divided into two categories. Most works regard this task as a deterministic generation problem. Zhao et al. [8] adopted a two-stage framework, i.e., generation and refinement. It first produced high-level spatial structure sequences, e.g., 2D positions of joints for a human pose, and then refined them with temporal visual features to generate fluent video frames. Although combining multi-level information, its spatial structure depends on a specific task (e.g., 3D Morphable Model for facial expression retargeting and 2D joints for human pose forecasting) and is not universal. Similarly, Yan et al. [11] adopted the object landmark as an intermediate structure before generating high-dimensional video data. They first used a recurrent model to get landmark sequences according to a motion category

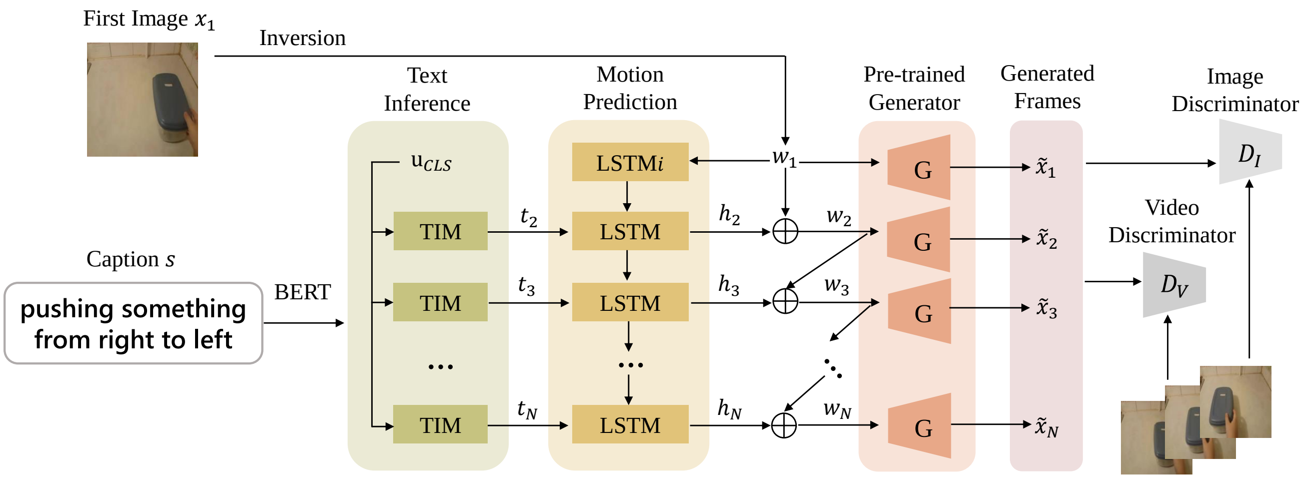

Fig. 2. The overview of the proposed Text-driven Video Prediction (TVP) framework. The text inference contains Text Inference Module (TIM) to explore the causality in the text description on motion evolution, generating step-wise inference embeddings t_n for each step, i.e., each frame. Then the motion prediction transforms them into motion residuals h_n to further construct latent codes w_n in a space shared by the pre-trained generator. Finally, the generator converts the latent codes w_n into predicted frames.

followed by landmark sequences to video network. Instead of making use of different structures, Li et al. [20] utilized a general form, optical flow, enabling flow prediction and flow-to-video synthesis. By contrast, DTVNet [21] mapped optical flow into a normalized motion vector and then integrated it with content features to generate videos. Rather than directly giving the first frame, Pan et al. [22] conditioned on semantic label map. They transformed it into the starting frame followed by a flow prediction network. Apart from the first frame, Blattmann et al. [23] inputted an extra poking of a pixel to enable interactive image-to-video synthesis, showing how local poke influenced the remainder of the object. Correspondingly, they designed a hierarchical recurrent network to predict fine-grained and complicated object dynamics. However, a single image is far from determining the whole video. Unlike the above works, others view image-to-video generation as a stochastic issue. For example, Dorkenwald et al. [9] suggested a bijective mapping between the video domain and static content provided by the first image, as well as other dynamic residual information sampled from the standard normal distribution. And they devised a conditional invertible neural network (cINN) to model such a relationship.

Different from the above two generation tasks, our proposed TVP provides the first frame and text description as strong constraints on appearance and motion factors to enable deterministic and controllable video generation. Although image-to-video generation is sometimes viewed as a similar setting, a frame is not able to decide the whole video. Therefore, TVP could be regarded as a more controllable and plausible task. The concurrent work [24] proposes a similar setting and adopts a VQ-VAE based framework, which is totally different from our GAN-based inference network. Besides, this work [24] uses the first image and text to produce a motion anchor, thus guiding the generation for all subsequent frames. In contrast, our framework fully explores the inference ability of text on motion information to generate step-wise embeddings specifically for each subsequent frame.

III. Methodology

A. Task Formulation

Considering that the current video generation tasks (i.e., unconditional and conditional video generation) lack constraints on both appearance and motion components to predicted frames, we propose a new task called Text-driven Video Prediction (TVP). Given the first frame x_1 and the text s describing motion changes of objects in x_1, TVP aims to generate the following frames $\{\hat{x}_2, \hat{x}_3, ..., \hat{x}_N\}$ accordingly, thus constructing the complete video $\hat{v}$. Especially, $\hat{v}$ tends to be the same as the ground-truth video v, fulfilling a deterministic and controllable video generation.

B. Motion Inference

1) Text Inference: For the TVP task, the appearance and motion factors are provided by the first frame x_1 and the text s respectively. And the motion in predicted frames is directly influenced by text. Therefore, the key to addressing the TVP task lies in mining the causality of text for motion information to generate semantically consistent videos. To investigate the controllable role of text descriptions for motion prediction in videos, we specially design the text inference G_T with the Text Inference Module (TIM). Given a sentence s with the length of M, pre-trained BERT [25] is adopted to obtain word-level embeddings $\{u_1, u_2, ..., u_M\}$. Furthermore, we hypothesize that each word in the sentence weighs differently for progressive motion inference. In other words, words in step-wise guidance for motion prediction have distinct importance and should be treated differently. To model such discrepancy among steps, we first calculate primary fusion representations f_i of word embeddings especially for each step i ($i = 2, 3, ..., N$):

$$f_i = \Sigma_{j=1}^{M} \alpha_{i,j} u_j, \tag{1}$$

$$\alpha_{i,j} = \frac{exp(W_i u_j)}{\Sigma_{k=1}^{M} exp(W_i u_k)}. \tag{2}$$

W_i denotes the step-wise and trainable weight. However, since such fusion embeddings f_i are produced independently for steps, it contains no global information, leading to incoherent construction for motion evolution across frames. To provide the guidance for overall coherency, we then make use of the sentence-level embedding u_{CLS}, i.e., the representations corresponding to [CLS] token, to refine f_i, thus combining global motion semantics. The refinement operation performed on primary fusion representations is calculated as follows:

$$t_i = P([f_i, u_{CLS}]). \tag{3}$$

In the above equation, P is a 2-layer MLP regarded as the transformation function. Thus, we get the step-wise inference embeddings $\{t_2, t_3, ..., t_N\}$ for motion prediction.

2) Motion Prediction: The motion prediction G_M incorporates the latent code $w_1 \in \mathbb{W}^+$ of the first frame and step-wise embeddings $\{t_2, t_3, ..., t_N\}$ to generate consecutive latent codes $\{w_2, w_3, ..., w_N\}$ for the generator to produce the following frames. Considering the architecture of motion prediction G_M, we utilize LSTM for its wide use and memory-friendliness. Moreover, the latent space $\mathbb{W}^+$ is shared by the fixed image generator StyleGAN [26], where the latent vector w is fed into each of the generator's layers via AdaIN [27] operation. The details of motion prediction are illustrated as follows.

First, to obtain the latent code $w_1 \in \mathbb{W}^+$ of the first frame, we make use of the existing GAN inversion model called FDIT [28], which aims to invert an image back to the latent space of a pre-trained generator. Then w_1 is used for initial cell $LSTM_i$ to get the initialization state of motion prediction:

$$h_1, c_1 = LSTM_i(w_1), \tag{4}$$

where h_1 and c_1 represent the hidden state and cell state respectively. Finally, the motion prediction G_M recursively takes step-wise embeddings $\{t_2, t_3, ..., t_N\}$ as inputs and produces hidden states $\{h_2, h_3, ..., h_N\}$ as latent residuals representing motion information to further attain codes $\{w_2, w_3, ..., w_N\}$ for the generator:

$$h_n, c_n = LSTM(t_n, (h_{n-1}, c_{n-1})), n = 2, 3, ..., N, \tag{5}$$

$$w_n = w_{n-1} + h_n. \tag{6}$$

More specifically, the hidden state h_n serves as motion residual between frame $n-1$ and n in the latent space $\mathbb{W}^+$. Therefore, adding h_n to the latent code of the last frame w_{n-1} could obtain the current one w_n, thus facilitating the generator to transform the latent code w_n into the predicted frame $\hat{x}_n$. Furthermore, the way to model motion information as latent residuals could largely maintain the appearance component.

C. Video Generation

1) Generator: For the image generator, we adopt the widely used StyleGAN [26]. Since GAN inversion operates on pre-trained GAN models, we first use our training datasets to get the generator G and fix it during the following process. Then the pre-trained generator G converts latent codes $\{w_2, w_3, ..., w_N\}$ in the latent space $\mathbb{W}^+$ to predicted frames $\{\hat{x}_1, \hat{x}_2, ..., \hat{x}_N\}$:

$$\hat{x}_n = G(w_n), n = 1, 2, ..., N. \tag{7}$$

2) Discriminator: To promote the generation results from the perspectives of 2D and 3D considering both spatial and temporal dimensions for predicted video frames, we adopt image discriminator D_I and video discriminator D_V. PatchGAN [29] is used as the base architecture for the two discriminators. As PatchGAN is designed for image modality, we replace its 2D convolution with 3D convolution to facilitate video modality. Furthermore, as suggested in [7], since the first predicted frame $\hat{x}_1$ is obtained by GAN inversion and thus in line with the distribution of the fixed image generator, the following ones should be conditioned on it. Therefore, for the input of image discriminator, we concatenate the RGB channel of the first frame $\hat{x}_1$ (x_1) and any subsequent frame $\hat{x}_n$ (x_n) to assess image-level quality. Similarly, for that of video discriminator, the RGB channel of $\hat{x}_1$ (x_1) are concatenate with all the following ones $\{\hat{x}_2, \hat{x}_3, ..., \hat{x}_N\}$ $(\{x_2, x_3, ..., x_N\})$ for evaluating the video quality.

D. Training

1) Generation Loss: To train G_T and G_M, we combine image-level loss and adversarial loss. For image-level loss, MSE and Perceptual losses considering pixel-level and high-level semantic similarities are adopted.

$$L_{mse} = ||v - \hat{v}||_2^2, \tag{8}$$

$$L_{perc} = \Sigma_p \frac{||\theta_p(x) - \theta_p(\hat{x})||_1}{C_p \times H_p \times W_p}. \tag{9}$$

$\theta_p(\cdot)$ represents the p-th layer of the pre-trained VGG-19 [30] producing $C_p \times H_p \times W_p$ feature maps. For adversarial training loss, 2D and 3D losses are calculated as follows, which aim to fool the discriminator to regard the generated results as the real ones from both image and video levels:

$$L_{2D} = logD_I(\hat{x}), \tag{10}$$

$$L_{3D} = logD_V(\hat{v}). \tag{11}$$

Finally, we combine the above losses with different weights to obtain the overall loss:

$$L = \lambda_{mse} L_{mse} + \lambda_{perc} L_{perc} + \lambda_{2D} L_{2D} + \lambda_{3D} L_{3D} \tag{12}$$

2) Discrimination Loss: To train D_I and D_V, we use the following losses separately, which target at promoting the discriminators to correctly distinguish the ground-truth videos and predicted outcomes:

$$L_{D_I} = logD_I(x) + log(1 - D_I(\hat{x})), \tag{13}$$

$$L_{D_V} = logD_V(v) + log(1 - D_V(\hat{v})). \tag{14}$$

IV. Experiments

A. Experimental Settings

1) Datasets: We evaluate our framework on two datasets: **Something-Something V2** [16] is a realistic dataset, consisting of 220,847 videos across 174 basic action classes. And each video ranges from 2 to 6 seconds. In our experiments, we select 8 categories with a larger number of videos: putting something on a surface, moving something up, pushing something from left to right, moving something down, pushing

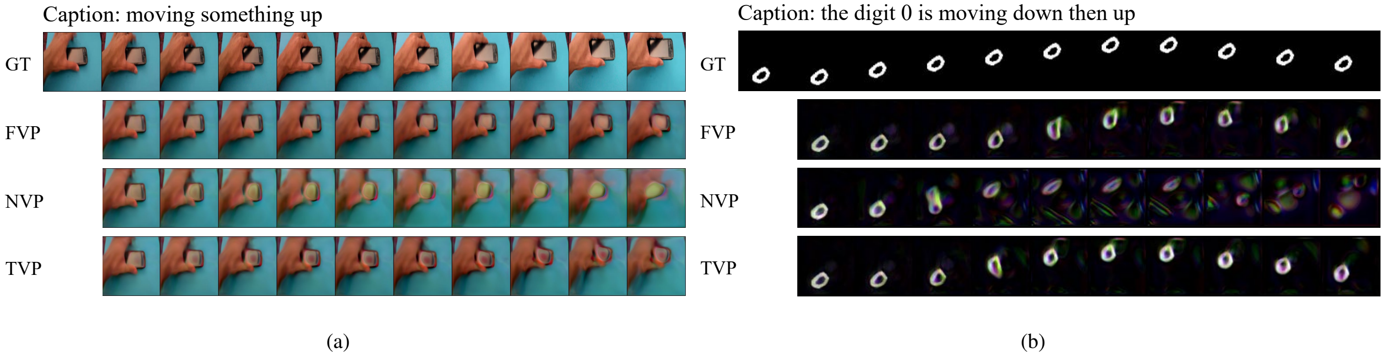
Caption: moving something up
GT
FVP
NVP
TVP
(a)
Caption: the digit 0 is moving down then up
GT
FVP
NVP
TVP
(b)

Fig. 3. Performance comparison over FVP and NVP baselines. Left: an example of Something-Something V2 dataset. Right: an example of Single Moving MNIST dataset.

something from right to left, squeezing something, pushing something so that it slightly moves and tearing something into two pieces. Moreover, these classes contain directional motions, making our TVP as deterministic video generation more reasonable. Regarding data split, we randomly sample 1500 and 100 videos from official sets as training and validation respectively for each category. Finally, we construct a dataset for our TVP with 12,000 training and 800 validation videos in total. Moreover, concerning complicated object categories as well as appearance and motion disentanglement, we directly adopt action labels as captions to form video-text pairs. **Single Moving MNIST** [15] is a synthetic dataset based on MNIST. It allows digits to follow four motion patterns, i.e., up then down, left then right, down then up and right then left, thus generating corresponding videos and captions. Furthermore, 10,000 and 2,000 videos are used for training and validation sets separately.

2) Performance Metrics: For evaluation, we adopt the metrics in video prediction task: Mean-Squared Error (MSE), Structural Similarity Index Measure (SSIM), Peak Signal to Noise Ratio (PSNR) and Learned Perceptual Image Patch Similarity (LPIPS). Specifically, MSE and PSNR focus on pixel-wise similarity between predicted and ground-truth frames, while SSIM takes image structural discrepancy into consideration. By contrast, LPIPS performs a semantic level comparison, as it calculates the difference of deep features. Moreover, higher SSIM and PSNR, lower MSE and LPIPS indicate better generation performance.

3) Implementation Details: Before training the proposed model, we conduct data pre-processing for videos and captions. For videos, we uniformly sample 11 (i.e., N=11) frames with different resolutions for two datasets, i.e., 256 × 256 for Something-Something V2 and 64×64 for Single Moving MNIST. Especially, although MNIST dataset contains gray images, we convert them into RGB frames, facilitating pre-training for StyleGAN. Regarding captions, maximum text length M is set to be 40. After that, StyleGAN is pre-trained for two datasets. Correspondingly, Something-Something V2 is used for a 256 × 256 StyleGAN with 14 layers and $w^+ \in \mathbb{R}^{14\times512}$, while Single Moving MNIST trains on a 64×64 StyleGAN with 10 layers and $w^+ \in \mathbb{R}^{10\times512}$. In Equation 12, λ_{mse}, λ_{2D} and λ_{3D} are 100, 1, 1 respectively for both datasets. As for λ_{perc}, we use 10 for Something-Something V2 while 1 is set for Single Moving MNIST. During training our TVP framework, with regard to the times to balance the generator (G_T and G_M) and discriminator (D_I and D_V), when the generator is to be trained twice, the discriminator will be trained once. Furthermore, Adam optimizer is utilized with a leaning rate of 0.0001 for G_T, G_M, D_I and D_V.

B. Compared Baselines

Since TVP is a new task, there are no directly comparable baselines. Considering that the novelty of our proposed TVP lies in verifying the causal inference capability of text for motion information, we design the following two baselines. **(1) Frame-only Video Prediction (FVP)**: FVP is the same as video prediction task conditioned on the first frame. Correspondingly, for the framework in Fig. 2, we remove the text inference and use the previous latent code w as the input of motion prediction to obtain the next code for the fixed image generator. **(2) Noise-driven Video Prediction (NVP)**: NVP is similar to the setting of image-to-video generation, where the input is the first image only and latent codes are sampled from distributions to provide motion information. Thus, we follow their common practice as we replace text inputs with noises. Concretely, we modify the t for motion prediction in Fig. 2 to noise variables in same feature dimensions. Moreover, the noise vectors are in line with standard normal distribution.

We compare our TVP framework with the above two baselines in Fig. 3. As shown in Fig. 3(a) for Something-Something V2 dataset, although the baseline FVP could generate the high-quality frames, the generated frames seem the same as the first frame without motion change. The results suggest giving only one frame is far from inferring subsequent frames. When incorporating random noises as in NVP, image quality is getting worse with no variation for motion, illustrating that noises not only couldn't be transformed into motion residuals but also entangle with the appearance factor. Therefore, noise variables have no capability to infer motion steps. Different from the above two baselines, our TVP captures the subtle motion change as "moving up" guide as well as maintains relatively better image quality. Moreover, the motion variation is almost synchronous to that in ground-truth frames, verifying the causal inference ability of text for motion information.

TABLE I
PERFORMANCE COMPARISON ON SOMETHING-SOMETHING V2 DATASET. METRICS ARE CALCULATED ON FRAME LEVEL.

Method	MSE↓	SSIM↑	PSNR↑	LPIPS↓
FVP	7186.5	**0.533**	**15.52**	**0.459**
NVP	7782.5	0.516	14.938	0.523
TVP w/o SE	7479.9	0.52	15.252	0.49
TVP w/o RM	7290.8	0.523	15.359	0.484
TVP	**7162.1**	**0.527**	**15.445**	**0.475**

TABLE II
PERFORMANCE COMPARISON ON SINGLE MOVING MNIST DATASET. METRICS ARE CALCULATED ON FRAME LEVEL.

Method	MSE↓	SSIM↑	PSNR↑	LPIPS↓
FVP	287.1	0.434	16.739	0.31
NVP	295.5	0.324	16.512	0.407
TVP w/o SE	264.6	0.426	17.032	0.313
TVP w/o RM	260.9	0.442	17.096	0.309
TVP	**255.8**	**0.455**	**17.239**	**0.299**

Similar observations can be also found in Fig. 3(b) for Single Moving MNIST dataset. When only inputting the first frame, the digit generated by FVP model firstly remains unchanged followed by moving upwards with appearance quality reduction. Although FVP could follow motion for videos containing simple appearance and motion as Single Moving MNIST dataset, it fails to achieve the correct trajectory. For NVP, there is no appearance and motion in predicted frames. It seems that noises play a dominant role and lead to a worse result. In contrast, our TVP framework achieves superior performance over two baselines as it correctly models the trajectory for digit and largely retains the appearance. Thus our framework fully explores the capability of text for providing motion inference. Overall, the above analysis of predicted examples in two datasets shows the advantage of our proposed TVP task and corresponding framework for deterministic video generation.

We further make quantitative comparisons between our TVP and baselines in Tables I and II. On Something-Something V2 dataset (Table I), FVP model achieves relatively better results, obtaining 7186.5, 0.533, 15.52 and 0.459 for MSE, SSIM, PSNR and LPIPS respectively. And our TVP gains comparable outcome on MSE, i.e., 7162.1 while other metrics are slightly lower than that of FVP, acquiring 0.527, 15.445 and 0.475 for SSIM, PSNR and LPIPS separately. Since NVP incorporating noises models incomplete motion information and intertwines two components (i.e., appearance and motion), it gets the worst results, i.e., 7782.5, 0.516, 14.938 and 0.523 on four metrics. For Single Moving MNIST dataset in Table II, similarly, NVP also gains the lowest statistics. Then as FVP maintains appearance and captures tiny motion, generating results surpass that of NVP, e.g., improving 0.11 for SSIM. In consideration of superior capability on motion inference of our TVP, it obtains the best performance on all four evaluation metrics, i.e., 255.8, 0.455, 17.239 and 0.299 on MSE, SSIM, PSNR and LPIPS respectively. Compared with FVP, TVP greatly enhances MSE metric for 31.3. It is worthwhile noting that FVP outperforms TVP quantitatively in Something-Something V2 dataset, while opposite conclusion could be found in Single Moving MNIST dataset. We speculate that the reason is due to the distinct properties of the two datasets. Concretely, since the objects in Something-Something V2 dominate the images, appearance plays a more significant role than motion. Considering FVP maintains better appearance than TVP, it achieves superior performance on evaluation metrics. By contrast, for Single Moving MNIST, frames contain small digits as well as simple background, resulting in sensibility for motion. Therefore, TVP receives best results for excellent modeling ability on both appearance and motion. Overall, from quantitative analysis, TVP shows its effectiveness in appearance preservation and motion inference for better video generation.

C. Ablation Study

The core ingredient of our proposed TVP framework is the Text Inference Module (TIM), which investigates the capability of text input to progressively infer motion information for video frames. To verify its effectiveness, we conduct ablation studies. Correspondingly, two variants are designed and demonstrated as follows. **(1) TVP w/o SE**: as described in Section 3.2, we produce step-wise embeddings (SE) $\{t_2, t_3, ..., t_N\}$ to specifically achieve motion inference for each frame. This variant replaces SE as sentence-level embeddings u_{CLS} for all following steps. **(2) TVP w/o RM**: in TIM, a refinement mechanism (RM) transforms primary fusion embeddings $\{f_2, f_3, ..., f_N\}$ to step-wise ones via incorporating global semantics thus guaranteeing coherent frame generation. And the second variant removes the RM. The comparison results are listed in Tables I and II for Something-Something V2 and Single Moving MNIST datasets separately.

In Table I on Something-Something V2 dataset, TVP w/o SE incorporates only global information for each prediction step and obtains 7479.9, 0.52, 15.252 and 0.49 on MSE, SSIM, PSNR and LPIPS metrics respectively. Compared with TVP w/o SE, TVP w/o RM inputs primary text fusion embeddings to motion prediction, possessing certain inference ability. Thus, it significantly enhances all four evaluation metrics. For example, MSE is raised from 7479.9 to 7290.8 and 0.107 is improved for PSNR metric. Although TVP w/o RM achieves better performance, this framework performs independent reasoning for each step and lacks global semantics. Therefore, a refinement mechanism is introduced in TVP to inject sentence-level semantics and insure coherent motion prediction. Moreover, TVP further boosts the generation results, obtaining the best metrics over two variants. For example, compared with TVP w/o RM, TVP increases PSNR from 15.359 to 15.445 and MSE metric is enhanced for 128.7. For Single Moving MNIST dataset in Table II, the same conclusion could be drawn as Something-Something V2 dataset. Since TVP w/o SE simply adopts global information and doesn't explore the inference potentiality in text, it achieves the worst generation performance, i.e., 264.6, 0.426, 17.032 and 0.313 on MSE, SSIM, PSNR and LPIPS metrics separately. On the basis of TVP w/o SE, TVP w/o RM enhances MSE for 3.7 and gains 0.442 on SSIM metric. Finally, after constructing RM, our

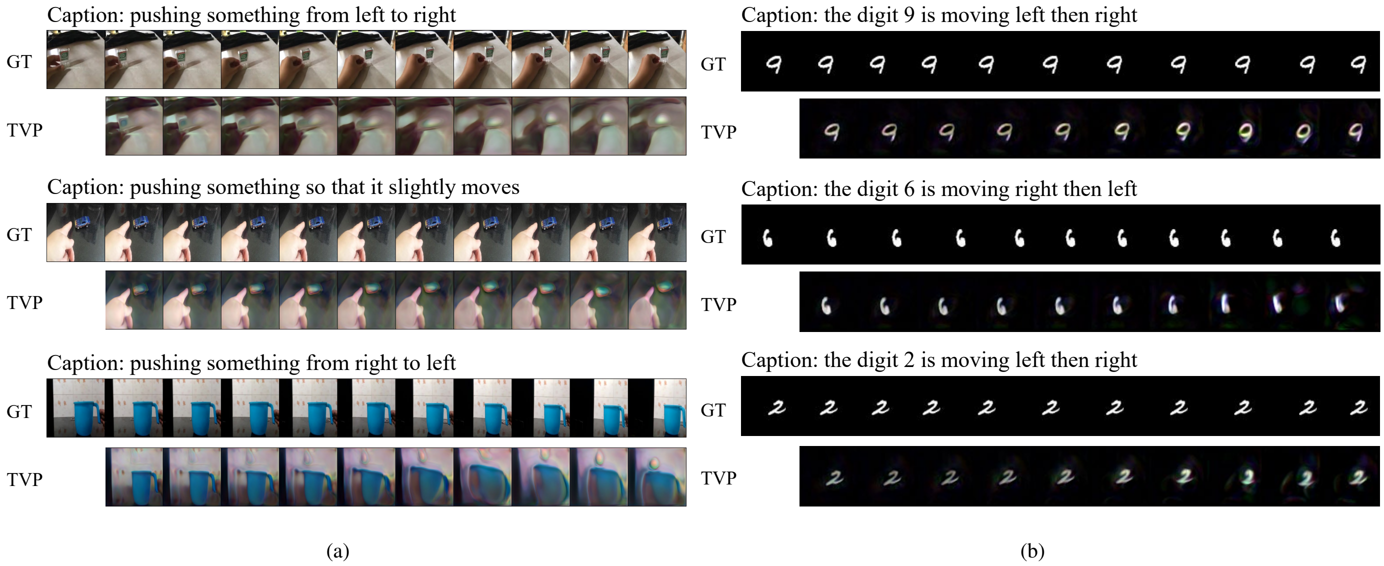
Caption: pushing something from left to right
GT
TVP
Caption: pushing something so that it slightly moves
GT
TVP
Caption: pushing something from right to left
GT
TVP
(a)
Caption: the digit 9 is moving left then right
GT
TVP
Caption: the digit 6 is moving right then left
GT
TVP
Caption: the digit 2 is moving left then right
GT
TVP
(b)

Fig. 4. Generated examples of TVP framework. Left: examples of Something-Something V2 dataset. Right: examples of Single Moving MNIST dataset.

TVP improves the performance to 255.8 on MSE and other metrics also give the best outcomes. Overall, considering the inference capability of text in motion, we correspondingly design the TIM for our TVP framework. Moreover, the ablation study completely verifies the efficiency of TIM, thus boosting the performance of TVP task.

D. Qualitative Results

In Fig. 4, we show more generated video examples conditioned on images and texts for both datasets. As shown in Fig. 4(a) on Something-Something V2, the first one describes “pushing something from left to right". In the ground-truth sequence, a bottle is obviously moved by a hand from left to right. With regard to the frames predicted by our TVP, although the appearance of hand and bottle becomes blurry, the corresponding motion trajectory is correctly captured. It could be seen that the positions of these two objects are matched in GT and TVP sequences. Different from this example, the second one shows “pushing something so that it slightly moves". And figures depict subtle action of the hand and toy car. Since this slight motion is difficult to observe, the positions of toy car could offer a good solution. From the comparison of two sequences, the position and even the shape of toy car are mainly aligned. It should be mentioned that the appearance discrepancy is largely caused by the loss of transforming the first image into the latent space w^+. The third example presents the opposite moving direction to the first one, i.e., “pushing something from right to left". In the ground-truth frames, the movement of the mug towards the left could be observed by the distance with the object in black color on the left even if the position of the mug is still on the right. Although our predicted sequence doesn’t capture such change, the mug follows the description in the caption and shows a proper motion trend. And it is moved by a hand to the left direction with a slight appearance decline. Considering the analysis of the above three examples, we conclude that despite the motion type, i.e., subtle or clear one to be noticed, our TVP seizes the difference in captions and generates correct motion correspondingly. As a result, the proposed TVP framework is capable to discover the causal inference in text and progressively produces aligned frames.

For Single Moving MNIST dataset in Fig. 4(b), we list examples showing different motion directions and digits to verify the advantages of our TVP framework for modeling both appearance and motion. Concerning the first example, given the text “the digit 9 is moving left then right" and the first image, our TVP largely maintains the digit’s appearance as well as depicts its trajectory. In the predicted video frames, there is a little inconsistency compared with GT sequences as the digit firstly moves left, while the later frames showing movement towards right are totally matched. In fact, achieving coherence of two motion directions in several frames is somehow difficult. Therefore, our TVP shows its deficiency in previous frame generation and finds its way later. For the second example, the caption gives opposite directions to the first one, i.e., “the digit 6 is moving right then left". Similarly, the appearance of digit 6 is preserved and the motion information mostly conforms to that in GT frames. With regard to the motion, the digit 6 in predicted frames shows little inclination towards right and then subsequently meets the left direction. The third example displays the same movement direction but a different digit compared with the first one. Similar conclusions could be drawn concerning motion tendency. It should be mentioned that the above three examples showing different digits convincingly demonstrate the excellent capability of our TVP framework for appearance preservation as we model motion information with residuals for subsequent frames conditioned on previous ones without appearance reduction.

Overall, from the examples of two datasets in Fig. 4, our TVP framework presents its superiority in motion prediction as captions guided and generates controllable and coherent frames as the real ones. Besides, it works better for single motion cases (either subtle or obvious motion) as in the Something-Something V2 dataset while representing weakness in more motion cases within several frames as for the Single Moving MNIST dataset.

V. Conclusion

To provide appearance and motion information for controllable and deterministic video generation, we propose a new task called Text-driven Video Prediction (TVP). It gives a static image and caption as inputs to generate subsequent video frames. Specifically, appearance and motion ingredients are offered by the image and text description respectively. As a cause-and-effect task, the main challenge is to investigate the causal inference ability in text. Correspondingly, we design a TVP framework as a base architecture to tackle this task. Moreover, the Text Inference Module (TIM) produces stepwise embeddings to infer motion change, thus facilitating progressive motion prediction. By making use of our framework, superior generation performance is achieved over baselines on Something-Something V2 and Single Moving MNIST.

Several issues are not been fully investigated in this paper and worth further exploration. Firstly, better results could be gained under multi-motion setting with more generated frames. As a result, how to utilize text to complete long-term motion inference and assignment while reducing appearance loss is a future direction. Secondly, since the classes we selected in Something-Something V2 and Single Moving MNIST contain up to two objects performing the same action, multi-object situation has not been explored. As this setting is more close to reality, it is another useful future direction. Overall, studying the TVP task under more complicated scenes is beneficial for real and advanced controllable video generation.

References

[1] R. Cui, Z. Cao, W. Pan, C. Zhang, and J. Wang, "Deep gesture video generation with learning on regions of interest," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2551–2563, 2019.

[2] K. Ohnishi, S. Yamamoto, Y. Ushiku, and T. Harada, "Hierarchical video generation from orthogonal information: Optical flow and texture," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[3] Y. Balaji, M. R. Min, B. Bai, R. Chellappa, and H. P. Graf, "Conditional gan with discriminative filter generation for text-to-video synthesis." in *IJCAI*, vol. 1, 2019, p. 2.

[4] Y. Wang, P. Bilinski, F. Bremond, and A. Dantcheva, "G3an: Disentangling appearance and motion for video generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5264–5273.

[5] W. Wang, X. Alameda-Pineda, D. Xu, E. Ricci, and N. Sebe, "Learning how to smile: Expression video generation with conditional adversarial recurrent nets," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2808–2819, 2020.

[6] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1526–1535.

[7] Y. Tian, J. Ren, M. Chai, K. Olszewski, X. Peng, D. N. Metaxas, and S. Tulyakov, "A good image generator is what you need for high-resolution video synthesis," *arXiv preprint arXiv:2104.15069*, 2021.

[8] L. Zhao, X. Peng, Y. Tian, M. Kapadia, and D. Metaxas, "Learning to forecast and refine residual motion for image-to-video generation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 387–403.

[9] M. Dorkenwald, T. Milbich, A. Blattmann, R. Rombach, K. G. Derpanis, and B. Ommer, "Stochastic image-to-video synthesis using cinns," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3742–3753.

[10] Y. Zhang, X.-Y. Liu, B. Wu, and A. Walid, "Video synthesis via transform-based tensor neural network," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2454–2462.

[11] Y. Yan, B. Ni, W. Zhang, J. Xu, and X. Yang, "Structure-constrained motion sequence generation," *IEEE Transactions on Multimedia*, vol. 21, no. 7, pp. 1799–1812, 2018.

[12] 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.

[13] Y. Li, M. Min, D. Shen, D. Carlson, and L. Carin, "Video generation from text," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[14] Y. Liu, X. Wang, Y. Yuan, and W. Zhu, "Cross-modal dual learning for sentence-to-video generation," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1239–1247.

[15] G. Mittal, T. Marwah, and V. N. Balasubramanian, "Sync-draw: Automatic video generation using deep recurrent attentive architectures," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1096–1104.

[16] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag *et al.*, "The" something something" video database for learning and evaluating visual common sense," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5842–5850.

[17] Y. Pan, Z. Qiu, T. Yao, H. Li, and T. Mei, "To create what you tell: Generating videos from captions," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1789–1798.

[18] K. Deng, T. Fei, X. Huang, and Y. Peng, "Irc-gan: Introspective recurrent convolutional gan for text-to-video generation." in *IJCAI*, 2019, pp. 2216–2222.

[19] T. Marwah, G. Mittal, and V. N. Balasubramanian, "Attentive semantic video generation using captions," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1426–1434.

[20] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang, "Flow-grounded spatial-temporal video prediction from still images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 600–615.

[21] J. Zhang, C. Xu, L. Liu, M. Wang, X. Wu, Y. Liu, and Y. Jiang, "Dtvnet: Dynamic time-lapse video generation via single still image," in *European Conference on Computer Vision*. Springer, 2020, pp. 300–315.

[22] J. Pan, C. Wang, X. Jia, J. Shao, L. Sheng, J. Yan, and X. Wang, "Video generation from single semantic label map," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3733–3742.

[23] A. Blattmann, T. Milbich, M. Dorkenwald, and B. Ommer, "Understanding object dynamics for interactive image-to-video synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5171–5181.

[24] Y. Hu, C. Luo, and Z. Chen, "Make it move: Controllable image-to-video generation with text descriptions," *arXiv preprint arXiv:2112.02815*, 2021.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186.

[26] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[27] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.

[28] M. Cai, H. Zhang, H. Huang, Q. Geng, Y. Li, and G. Huang, "Frequency domain image translation: More photo-realistic, better identity-preserving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 930–13 940.

[29] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.