Asia Pacific Academy of Science Pte. Ltd.

Metaverse 2024, 5(1), 2024.
https://doi.org/10.54517/m.v5i1.2425

Article

Text to video generation via knowledge distillation

Huasong Han, Ziqing Li, Fei Fang, Fei Luo, Chunxia Xiao*

Wuhan University, Wuhan 430072, China
*** Corresponding author:** Chunxia Xiao, cxxiao@whu.edu.cn

CITATION

Han H, Li Z, Fang F, et al. Text to video generation via knowledge distillation. Metaverse. 2024; 5(1): 2425.
https://doi.org/10.54517/m.v5i1.2425

ARTICLE INFO

Received: 13 December 2023
Accepted: 25 January 2024
Available online: 19 March 2024

Abstract: Text-to-video generation (T2V) has recently attracted more attention due to the wide application scenarios of video media. However, compared with the substantial advances in text-to-image generation (T2I), the research on T2V remains in its early stage. The difficulty mainly lies in maintaining the text-visual semantic consistency and the video temporal coherence. In this paper, we propose a novel distillation and translation GAN (DTGAN) to address these problems. First, we leverage knowledge distillation to guarantee semantic consistency. We distill text-visual mappings from a well-performing T2I teacher model and transfer it to our DTGAN. This knowledge serves as shared abstract features and high-level constraints for each frame in the generated videos. Second, we propose a novel visual recurrent unit (VRU) to achieve video temporal coherence. The VRU can generate frame sequences as well as process the temporal information across frames. It enables our generator to act as a multi-modal variant of the language model in neural machine translation task, which iteratively predicts the next frame based on the input text and the previously generated frames. We conduct experiments on two synthetic datasets (SBMG and TBMG) and one real-world dataset (MSVD). Qualitative and quantitative comparisons with state-of-the-art methods demonstrate that our DTGAN can generate results with better text-visual semantic consistency and temporal coherence.

Keywords: text-to-video; generation; knowledge; distillation; GANs; GRUs

1. Introduction

Text is the symbolic carrier of information and human thought, while visual content (images, videos, etc.) provides a way for humans to interact, understand, and learn about the world. In the human consciousness, there exists a semantic correspondence between text and visual content. Using deep learning to realize the mutual transformation between text and visual content [1–3] is of great research value.

In this paper, we focus on the text-to-video generation (T2V) task. Specifically, given a piece of input text, we aim to generate a video that is realistic, coherent, and semantically consistent with the input text, as shown in **Figure 1**. It is widely used in various fields such as multimedia teaching, social media, virtual reality, etc.

There are two key issues in this task: 1) text-visual semantic consistency and 2) temporal coherence across video frames. Several approaches have been proposed to address them. For the first issue, current T2V methods usually employ more loss constraints. For example, Balaji et al. [4] incorporated a text filter to the discriminators. Deng et al. [5] proposed a mutual-information introspection loss. However, the effect of loss constraints is limited. State-of-the-art T2I methods [6–8] have good capability in semantic consistency preservation while existing T2V works ignore the reference values of T2I methods.

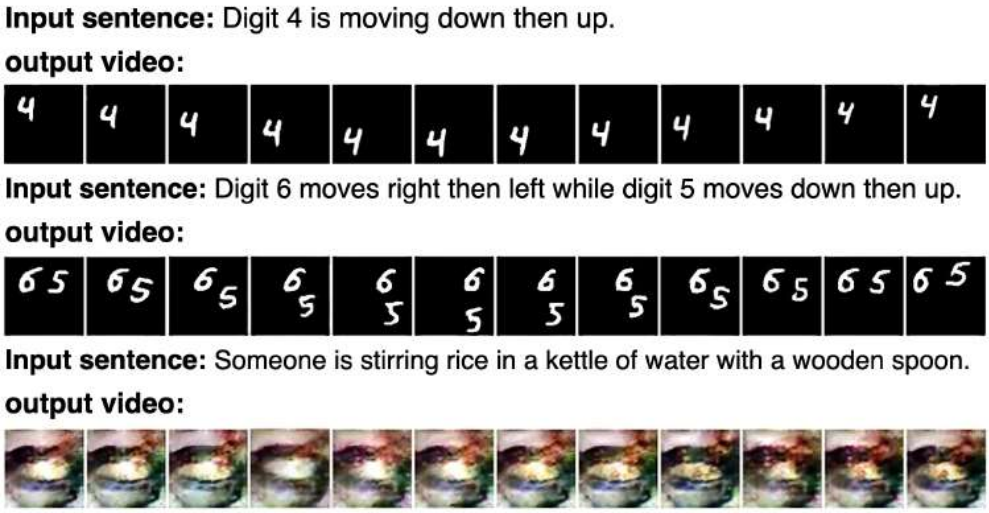

Figure 1. Examples of video generated by our DTGAN on single-digit bouncing MNIST GIFs, two-digit bouncing MNIST GIFs and Microsoft research video description corpus, respectively.

For the second issue, latest T2V methods made efforts in following aspects. For the generator, some works adopted 3D deconvolutional layers to capture the temporal information globally [9,10]; others [4,5,11] fused temporal process modules on the 2D deconvolutional layers to capture the temporal information in the latent space. For discrimination, works [4,9] adopted more discriminators that judge the results from video or motion perspective to improve the discriminative ability. However, existing methods fail to notice the seq2seq nature of T2V, whose solutions explore associations between sequence nodes and thus can help to improve the coherence across frames.

To tackle the above issues, we propose a novel distillation and translation GAN (DTGAN), as shown in **Figure 2**. First, we leverage knowledge distillation (KD) to enhance text-visual semantic consistency. We investigate well-performing T2I methods for semantic consistency solutions and find that these solutions rely on deep deconvolutional layers and complex internal designs. Directly adopting them would greatly increase the computational load, so we innovatively use KD to exploit well-performing T2I models indirectly. KD can transfer knowledge in model parameters from T2I model to T2V model, which serves as shared abstract features and high-level constraints for each video frame, thereby improving the realism and semantic consistency of the whole video. Second, we propose a novel visual recurrent unit (VRU) to resolve video temporal coherence. It is suitable to adopt RNN for T2V task because T2V is inherently a seq2seq problem. We refine GRU as VRU to enable it to generate visual content and capturing temporal information. With VRU, our generator works as a multi-modal variant of the neural machine translation (NMT) decoder [12,13] that iteratively predicts the next frame based on the input and previously generated frames (**Figure 3**).

In this way, we guarantee the temporal coherence of the generated frame sequences.

To our best knowledge, we are the first to apply KD in T2V, which leverages the advantages of T2I models to deal with the challenge problem in T2V.

Specifically, our work has the following contributions:

- We propose a teacher-aided semantic capture module that uses KD to transfer text-visual mapping knowledge from T2I model to T2V model. This helps to improve the semantic consistency of the generated videos.
- We exploit a conditional visual model generator (cVMG) that contains a novel VRU to iteratively generate video frames with temporal coherence.
- We provide quantitative and qualitative experimental comparisons to demonstrate the capability of our DT-GAN. We also conduct an ablation study to verify the effectiveness of each proposed component in our model.

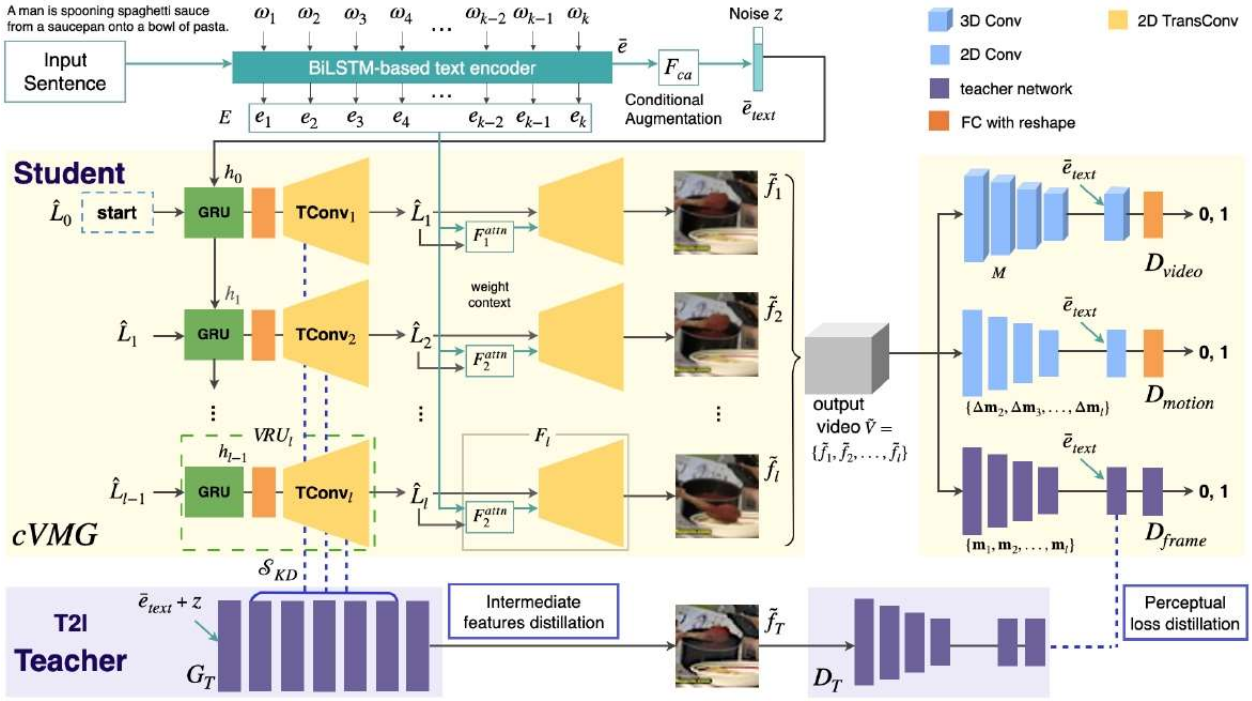

Figure 2. The overview of DTGAN. The VRU (green dotted box) enables our cVMG to iteratively translate text into frames. The KD mechanism (purple dotted line) provides shared abstract features and high-level constraints for each video frame.

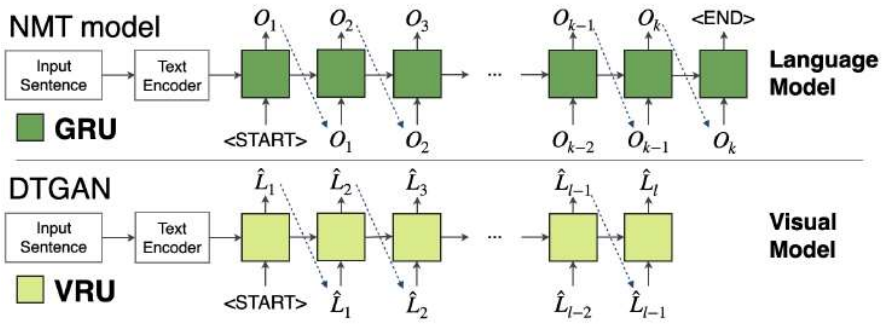

Figure 3. Comparison of NMT model and our DTGAN. The decoder of NMT is a language model that generates the target sentence "O1O2...Ok" conditioned on the input encoding. Since our result is of fixed length, the <END> node is excluded in our visual model.

2. Related work

A. Text to image generation:

Reed et al. [14] generated images from text by using GAN and implementing a text encoder that mapped text descriptions to the common feature space of the image. Based on this method, Zhang et al. [6] proposed StackGAN++ with stacked generators to generate higher resolution results. Xu et al. [7] proposed AttnGAN with the attention mechanism and a DAMSM to improve fine-grained details. Zhang et al. [8] employed hierarchically-nested discriminators at multi-scale intermediate layers of the generator to generate images of different resolutions. The generators of StackGAN++, AttnGAN, and HDGAN commonly employ deep deconvolutional layers and complex internal designs, which on the one hand, enable them to generate realistic single-object images (e.g., birds, flowers) that are semantically consistent with the input text, and on the other hand, bring a large computational load.

B. Text to video generation:

Early works used conditional GANs for T2V task. Pan et al. [9] adopted a 3D deconvolutional generator and three discriminators that judged the results from video, frame, and motion perspectives. Chen et al. [10] added attention mechanism to promote word-region level consistency. Kim et al. [15] generated one image first and then synthesized consecutive frames in further stages.

There are also works that generate static background and dynamic foreground separately [16,17].

Recent works integrated temporal processing modules into the generators to promote temporal coherence. For example, Deng et al. [5] proposed recurrent transconvolutional generator (RTG), where LSTM cells were integrated with 2D transconvolutional layers.

Mazaheri et al. [11] used linear interpolation to get the conditional latent representation for each frame. Balaji et al. [4] adopted a GRU in the latent space and a shared frame generator network similar to mocoGAN [18].

There are also works that take other strategies. Liu et al. [19] adopted a dual learning mechanism to learn the bidirectional mappings between input text and generated videos. In addition to generating videos from scratch, Gupta et al. [20] retrieved spatial-temporal entity segments from a video database and fused them to generate scene videos. Recently, some works adopted VQ-VAE to generate videos and achieved state-of-the-art performance [21,22].

C. Knowledge distillation:

Knowledge distillation (KD) works in a teacher-student manner. It was first proposed by Hinton et al. [23], who clarified that the idea of KD is to allow the student model to achieve certain competitive performance by imitating the teacher model. The chosen teacher networks usually have privileged information that student networks do not have. For example, teachers have a deeper neural network or have more input data.

We mainly review the works of KD involving GANs. There are three ways for transferring information between teacher and student.

1) Pix-to-pix loss. Aguinaldo et al. [24] first showed how to use KD in GANs, which directly used MSE or L1 loss to guide the student generator through

minimizing the euclidean distance between the synthesized images of teacher and student.

2) Perceptual loss [25]. Chen et al. [26] introduced it to GAN distillation, which used the teacher discriminator to measure the high-level distance between teachers and students as the perceptual loss.

3) Intermediate feature distillation. Li et al. [27] added a learnable convolutional layer between the intermediate layers of student and teacher generators to achieve intermediate feature distillation. Jin et al. [28] adopted kernel alignment to directly force the intermediate feature representations from the two models to be similar. Li et al. [29] proposed to transfer the attention maps of the intermediate representations, as they contained more details.

In our work, we innovatively use KD for transferring text-visual mapping knowledge to improve the generation capability of our model.

3. Method

Our DTGAN consists of four components: the text encoder network, the conditional visual model generator (cVMG), the discriminators, and the teacher-aided semantic capture module. We first introduce these four components and then introduce the objective function.

A. Text encoder:

We employ a bi-directional long short-term memory (Bi-LSTM) [12] text encoder to extract semantic vectors from the input text. First, the input text with k words can be represented as a set of one-hot vectors $\{w_1, w_2, ..., w_k\}$. Then, the vectors are fed into a Bi-LSTM encoder to get the contextually embedded word sequence $E = \{e_1, e_2, \ldots, e_k\}$. We treat the concatenation of the two last hidden states of the Bi-LSTM as the sentence vector $\overline{e}$. F_{ca} in **Figure 2** represents the conditioning augmentation [6] that converts the sentence vector $\overline{e}$ to the conditioning text latent code $\bar{e}_{text} \in R^{d_{text}}$.

B. Conditional visual model generator (cVMG):

As shown in **Figure 3**, we generate frames in a way similar to NMT decoders [12,13]. Since GRU cannot generate images, we propose a novel visual recurrent unit (VRU) that can both generate visual content and capture temporal information. By adopting VRU, our generator acts as a conditional visual model that translates input text into frame sequences.

Details of network: To generate a video with l frames, the cVMG has l timestep (**Figure 2**). Initially, we generate a random noise frame $\hat{L}_0 \in R^{d_c \times d_h \times d_w}$ to act as the START node and use the concatenation of $z \in R^{d_z} \sim \mathcal{N}(0,1)$ and $\bar{e}_{text}$ text to get the initial state h0 of the GRU. The proposed cVMG consists of l visual recurrent units $\{VRU_1, VRU_2, ..., VRU_l\}$ and l frame generation modules $\{F_1, F_2, ..., F_l\}$. The VRU comprises a GRU cell, a fully-connected layer, and a 2D transconvolutional layer TConv. The F consists of an attention module Fattn [7,10] and another 2D transconvolutional layer. Note that the attention module here helps to enhance word-region level fine-grained details.

At timestep *t*, the *V*RU*t* takes $\tilde{L}_{t-1}$ as input and generates raw image $\tilde{L}_t$. ht is updated by the GRU cell to represent the temporal information and the semantic features from input text. The raw image $\tilde{L}_t$ is then sent to F_t to generate the final output $\tilde{f}_t \in R^{d_c \times d_h \times d_w}$ at timestep t. Here d_c, d_h, d_w denote the frame channels number, frame height, and frame width, respectively. Specifically,

$$\begin{aligned} \hat{L}_t, h_t &= VRU_t(\hat{L}_{t-1}, h_{t-1}), \quad t = 1, 2, ..., l, \\ \tilde{f}_t &= F_t(\hat{L}_t, F_t^{attn}(E, \hat{L}_t)), \end{aligned} \tag{1}$$

where $E \in R^{d_{text} \times k}$ denotes the feature matrix of all words.

Finally, we put the generated frames $\{\tilde{f}_1, \tilde{f}_2, ..., \tilde{f}_l\}$ together to form a whole video $\tilde{V} \in R^{l \times d_c \times d_h \times d_w}$. The overall process is:

$$\tilde{V} = cVMG(\bar{e}_{text}, z). \tag{2}$$

C. Discriminator networks:

We use three discriminators to judge whether the output is real or fake from the following perspectives [9]: (1) the whole video, (2) the motion across adjacent frames, (3) each video frame.

- Video discriminator D_{video} captures the global information over the entire video. It uses a 3D convolutional neural network to extract video-level features M. Then, we augment M with corresponding text embedding $\overline{e}_{text}$ and feed them into a convolutional layer followed by a fully-connected layer with softmax to verify whether the video is semantically matched with the given text and whether it is fake or real.
- Motion discriminator Dmotion processes the input video in a frame-wise way. We first extract frame-level features mt for each frame using 2D convolutional layers. Next, we continuously calculate the motion between two consecutive frames and get the set of $\{\Delta \boldsymbol{m}_2, \Delta \boldsymbol{m}_3, \ldots \Delta \boldsymbol{m}_l\}$ to represent the information about the temporal coherence. Then we obtain frame motion discrimination through 2D convolutional layers and fully-connected layers.
- Frame discriminator D_{frame} does the same job as the teacher discriminator, so we adopt the teacher discriminator to act as D_{frame}.

D. Teacher-aided semantic capture module:

We empirically observe that deeper deconvolutional layers and complex internal designs (e.g., stacked structure) enable generators to generate higher-quality images. However, applying these to T2V generators is impractical due to the high computational cost. Therefore, we choose a well-performing T2I network containing the techniques mentioned above as the teacher and distill knowledge from it to aid the T2V model in improving the realism and text-visual semantic consistency of the generated videos. In this way, we do not increase the computational cost too much.

The key to using KD is how to measure the distance between the teacher and the student so that we can transfer knowledge by minimizing their gap. Since each generated frame has different motions, we do not use pixel-level direct knowledge transfer to avoid reducing the dynamics of the generated videos. The features of intermediate layers in the teacher network are useful as they contain rich information

about mappings from the text domain to the visual domain.

We use the intermediate features as shared abstract features for video frames. We directly encourage similarity between teacher and student feature space through kernel alignment (KA) [28,30–32]. We also transfer the attention map [29,33] of the intermediate layer parameters, as such metric can well concentrate information. Furthermore, we convey perceptual information by using the teacher discriminator as high-level guidance, which helps our generator to generate results with better text-visual semantic consistency and realism.

Intermediate feature distillation: As illustrated in **Figure 4**, the intermediate layers selected for distillation are denoted as $\mathcal{S}_{KD}$. We perform distillations between the teacher generator and TConv in our VRU cell at each timestep.

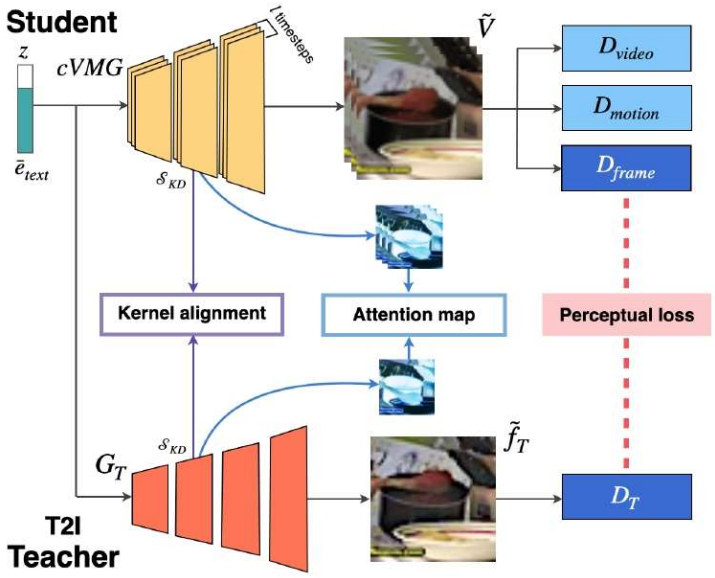

Figure 4. Teacher aids semantic capture module. We use kernel alignment (KA) and attention map to distill knowledge in the intermediate layers SKD. We use teacher discriminator to distill high-level features as perceptual loss.

At timestep *t*, we transfer attention maps and encourage a direct similarity between their feature space through KA for $TConv_t$. The corresponding objective function is:

$$L_{inter_t} = -\lambda_{KA} \sum_{p \in \mathcal{S}_{KD}} K\,A(G_T^{(p)}, TConv_t^{(p)}) + \lambda_{Att} \sum_{p \in \mathcal{S}_{KD}} \| \frac{A_T^{(p)}}{\|A_T^{(p)}\|_2} - \frac{A_{TConv_t}^{(p)}}{\|A_{TConv_t}^{(p)}\|_2} \|_2^2 \quad (3)$$

where the minus sign is introduced as we intend to maximize feature similarity between student and teacher models, λ_{KA} and λ_{Att} are pre-defined hyper-parameters. $G_T^{(p)}$ and $G_T^{(p)}$ denote features of layer p from the teacher and student network. $A_{G_T}^{(p)}$ and $A_{TConv_t}^{(p)}$ denote attention maps of layer p from the teacher and student network. $KA(\cdot)$ is defined following [28], and attention maps A are obtained following [33].

Finally, for a video clip with l frames, there are total l Tconv layers that need to be distilled. The overall objective function for intermediate feature distillation is:

$$\mathrm{L}_{inter} = \frac{1}{l}\sum_{t=1}^{l}\mathrm{L}_{inter_t} \tag{4}$$

Perceptual loss distillation: As illustrated in **Figure 4**, we use the teacher discriminator to extract features of images generated by the teacher and each synthesized frame as follows:

$$L_{perc_t} = \lambda_{perc}(\widehat{D}_T(\tilde{f}_T, \bar{e}_{text}) - \widehat{D}_T(\tilde{f}_t, \bar{e}_{text}))) \tag{5}$$

$$L_{perc} = \frac{1}{l}\sum_{t=1}^{l} L_{perc_t} \tag{6}$$

where λ_{perc} is a pre-defined hyperparameter, $\tilde{f}_T$ is the images generated by the teacher model conditioned on the input text, $\tilde{f}_t$ denotes the video frame generated at timestep *t*, and $\widehat{D}_T$ is the last convolutional layers of the discriminator of the teacher network. The teacher discriminator can effectively capture the manifold of the target domain.

Our frame discriminator D_{frame} inherits the architecture and the pre-trained weight from the teacher discriminator for 64 × 64 resolution images following [27]. We fine-tune the weight of D_{frame} in the training process. Such an approach avoids the instability problem with randomly initialized weights in the early stage of training.

E. Objective function:

According to the above discussion, the overall objective function of the discriminator networks is as follows:

$$\mathrm{L}_D = \mathrm{L}_{D_{video}} + \mathrm{L}_{D_{motion}} + \mathrm{L}_{D_{frame}}, \tag{7}$$

$$\begin{aligned}\mathrm{L}_{D_{video}} = -\frac{1}{3}[&\log D_{video}(V, \bar{e}_{text}) \\ &+ \log(1 - D_{video}(V', \bar{e}_{text})) \\ &+ \log(1 - D_{video}(\tilde{V}, \bar{e}_{text}))],\end{aligned} \tag{8}$$

$$\begin{aligned}\mathrm{L}_{D_{motion}} = -\frac{1}{3(l-1)}\Big[&\sum_{i=2}^{l}\log D_{motion}(\Delta\mathbf{m}_i, \bar{e}_{text}) \\ &+ \sum_{i=2}^{l}\log(1 - D_{motion}(\Delta\mathbf{m}_i', \bar{e}_{text})) \\ &+ \sum_{i=2}^{l}\log(1 - D_{motion}(\Delta\widetilde{\mathbf{m}}_i, \bar{e}_{text}))\Big],\end{aligned} \tag{9}$$

$$\begin{aligned}\mathrm{L}_{D_{frame}} = -\frac{1}{3l}\Big[&\sum_{i=1}^{l}\log D_{frame}(f_i, \bar{e}_{text}) \\ &+ \sum_{i=1}^{l}\log(1 - D_{frame}(f_i', \bar{e}_{text})) \\ &+ \sum_{i=1}^{l}\log(1 - D_{frame}(\tilde{f}_i, \bar{e}_{text}))\Big].\end{aligned} \tag{10}$$

Here V, V' , $\tilde{V}$ are real video, mismatched real video and our synthesized video, respectively. $\Delta\mathbf{m}_i, \Delta\mathbf{m}_i', \Delta\widetilde{\mathbf{m}}_i$ are the motion features between the ith and $(i-1)$-th

frames in video V, V' , $\mathcal{V}$, respectively. $f_i, f_i^{'}, \tilde{f}_i$ denote the ith frame in video V, V', $\mathcal{V}$, respectively.

The training objective for the generator network cVMG is:

$$\mathrm{L}_G = \mathrm{L}_{G_v} + \mathrm{L}_{G_m} + \mathrm{L}_{G_f} + \mathrm{L}_{DAMSM} + \mathrm{L}_{inter} + \mathrm{L}_{perc}, \tag{11}$$

where L_{inter} and L_{perc} are defined in Equations (4) and (6). L_{DAMSM} is the DAMSM loss [7] that enhances word-region level finegrained consistency. L_{G_v} , L_{G_m}, L_{G_f} are objective functions of typical GANs, which are defined as:

$$\mathrm{L}_{G_v} = -\frac{1}{3}\log(D_{video}(\tilde{V}, \overline{e}_{text})), \tag{12}$$

$$\mathrm{L}_{G_m} = -\frac{1}{3(l-1)}\sum_{i=2}^{l}\log(D_{motion}(\Delta\widetilde{\mathbf{m}}_i, \overline{e}_{text})), \tag{13}$$

$$\mathrm{L}_{G_f} = -\frac{1}{3l}\sum_{i=1}^{l}\log(D_{frame}(\tilde{f}_i, \overline{e}_{text})). \tag{14}$$

Note that pre-defined hyper-parameters λ_{KA} , λ_{Att} , and λ_{perc} are used in Equations (3) and (5) to balance the objective.

4. Experiment

A. Datasets:

SBMG. We adopt the modified version of Single-Digit Bouncing MNIST GIFs [5]. It is automatically generated from MNIST dataset by having a 28 × 28 single handwritten digit bouncing inside a 64 × 64 frame. It is composed of 12,000 GIFs of 16 frames long.

TBMG. Two-Digit Bouncing MNIST GIFs [5] is an extension of SBMG with two handwritten digits bouncing. The generation process is the same as SBMG and the two digits move separately.

MSVD. Microsoft Research Video Description Corpus is a popular video captioning benchmark of YouTube videos. It contains 1970 video snippets with roughly 40 available English descriptions per video. We follow [9,10] to use the subset of 421 cooking videos, with 361 videos for training and 60 for testing.

B. Experimental settings:

Parameter settings. Each video clip has l = 16 frames. The hyper-parameters λ_{perc}, λ_{KA}, and λ_{Att} are set to 0.0001, 0.7, 60 for SBMG and TBMG, and 0.0001, 0.6, 80 for MSVD, respectively. The learning rate is kept to 0.0002 in the beginning and linearly decayed to zero. For all experiments, we adopt the Adam optimizer with a tuple of beta values as (0.9, 0.999) for both generator and discriminator.

Implementation details. We train each model on the GPUs of GeForce RTX 2080Ti with a memory capacity of 11 GB. Due to limitations in GPU memory in the laboratory, it is not possible to generate longer video frames. Therefore, we generate videos with resolution 64 × 64 and 16 frames.

For sentence encoding, the dimension of the input, hidden layers, output in bi-LSTM are all set to 256. The dimension of the random noise variable z, i.e., d_Z is 100. All weights were initialized from a zero-centered Normal distribution with

standard deviation 0.02.

Evaluation metrics. We use FID [34] to evaluate the quality of a single frame. It measures the distances between real frames and the generated frames in the feature space. We also adopt FID2vid to measure both quality and temporal consistency of the whole video. A lower FID score or FID2vid score denotes a better quality of the generated results.

We adopt the Generative Adversarial Metric (GAM) to directly compare two generative models by making them engage in a "battle" against each other. Given two generative adversarial models $M_1 = \{(\tilde{G}_1, \tilde{D}_1)\}$ and $M_2 = \{(\tilde{G}_2, \tilde{D}_2)\}$, the ratios are defined as:

$$r_{test} = \frac{\grave{o}(\tilde{D}_1(x_{test}))}{\grave{o}(\tilde{D}_2(x_{test}))}, r_{sample} = \frac{\grave{o}(\tilde{D}_1(\tilde{G}_2(x)))}{\grave{o}(\tilde{D}_2(\tilde{G}_1(x)))}, \tag{15}$$

If r_{test} is close to 1, it means the two models have almost the same ability to recognize the real videos. Then if sample < 1, it means that G1 can fool D2 more easily (noted that this method is restricted for GANs, so VAE-based methods (e.g., Sync-DRAW [35]) are excluded from this comparison).

C. Teacher network

We choose AttnGAN [7] as our teacher network, which can generate single-object images (e.g., birds, flowers) with high quality. Under the premise of maintaining the performance of AttnGAN, we slightly modify the realization of deconvolutional layers in the generators of AttnGAN to make it consistent with our TConv layers. Note that the generator of AttnGAN has seven deconvolutional layers and can generate images with resolutions 64 × 64, 128 × 128, and 256 × 256.

D. Quantitative and qualitative evaluations

1) Quantitative comparison via FID and FID2vid: As shown in **Table 1**, our DTGAN outperforms existing state-of-the-art methods. We get the best FID score, which means our generated videos have the best visual quality. The visual performance improvement over BoGAN and TFGAN is mainly attributed to the knowledge distilled from the teacher model. At the same time, our DTGAN also gets the best FID2vid score. The FID2vid score of TFGAN is inferior to DTGAN. Although TFGAN and DTGAN both integrate recurrent networks, our VRU is more suitable for T2V task, so it can generate more coherent videos. The overall results indicate that our DTGAN is capable of producing videos which not only contain realistic frames but also have temporal coherence over frame sequence.

Table 1. Quantitative comparison via fid and fid2vid with state-of-the-art methods (64 × 64).

Method	**SBMG**		**TBMG**		**MSVD**	
	FID	FID2vid	FID	FID2vid	FID	FID2vid
Sync-DRAW [35]	72.33	4.77	121.62	4.93	287.52	18.45
TGANs-C [9]	67.67	4.59	61.77	5.20	192.44	14.85
TFGAN [4]	40.76	298	38.92	3.55	103.03	9.71
BoGAN [10]	7.57	3.12	48.31	4.22	74.55	9.85
DTGAN (ours)	**28.78**	**2.73**	**31.63**	**3.15**	**70.92**	**8.87**

2) Quantitative comparison via GAM: For evaluation on GAM, we choose a more complex dataset MSVD for comparison, which can better reflect the performance of the model. From **Table 2**, the rsample scores are all less than one, indicating that our DTGAN can generate videos that fool the discriminators of other GAN-based methods. In other words, compared with the videos generated by other methods, ours are more realistic, more coherent, and more consistent with the text.

Table 2. Quantitative comparison with gam metric on the MSVD dataset. When $r_{test} \approx 1, r_{sample} < 1$ means the former beats the latter.

Battler	r_{test}	r_{sample}	**Winner**
DTGAN vs. TGANs-C [9]	1.07	0.55	DTGAN
DTGAN vs. TFGAN [4]	0.98	0.91	DTGAN
DTGAN vs. BoGAN [10]	0.97	0.83	DTGAN

3) Qualitative comparison: We provide a qualitative evaluation by comparing our DTGAN with GAN-based approaches [4,5,9,10], VAE-based approach [35], and VQVAE-based approach [21]. **Figure 5a,b** shows the results on SBMG dataset and TBMG dataset, respectively. It can be observed that digits generated by VAE (Sync-DRAW) or 3D deconvolutional layer based generator (TGANs-C, BoGAN) still have slight motions in the unmentioned direction. Our model and state-of-the-art IRC-GAN and GODIVA models can all generate digit 9, 3 and 7 with correct motion and more stable shape even when the two digits 3 and 7 cluster together, while other earlier methods have slight distortions in several frames. Nevertheless, our model has lighter computational loads than IRC-GAN and GODIVA. **Figure 5c** shows the results on the MSVD. (We do not show the results of IRCGAN [5] and GODIVA [21] on the MSVD dataset, because they didn't provide results on this dataset and their source code was not released). The videos generated by our DTGAN have clearer edges and less noise, as well as reasonable dynamics conditioned on the input text. The overall qualitative results demonstrate that our model achieves state-of-the-art performance.

E. Ablation study:

We construct the baseline model by removing KD loss in the DTGAN model and replacing VRU in the generator with a simple GRU that only captures temporal information in latent space. Then we add the proposed modules one by one to verify their effectiveness.

As shown in **Table 3**, the baseline model with these two retained modules are compared with our DTGAN via FID, FID2vid. The model with KD loss LKD has the worst FID2vid score (slightly lower than the baseline model). This is acceptable because KD loss can only improve the quality of each single frame. This model achieves the best FID score, which is a large improvement over the baseline model. It shows that KD loss has a significant effect on improving the image quality. The model with VRU achieves both improved FID score and FID2vid score compared to the baseline model. This means generator with our proposed module is more suitable for T2V task than a simple GRU. Finally, the whole model achieves the best or near-best scores on both metrics, which demonstrates the effectiveness of our method.

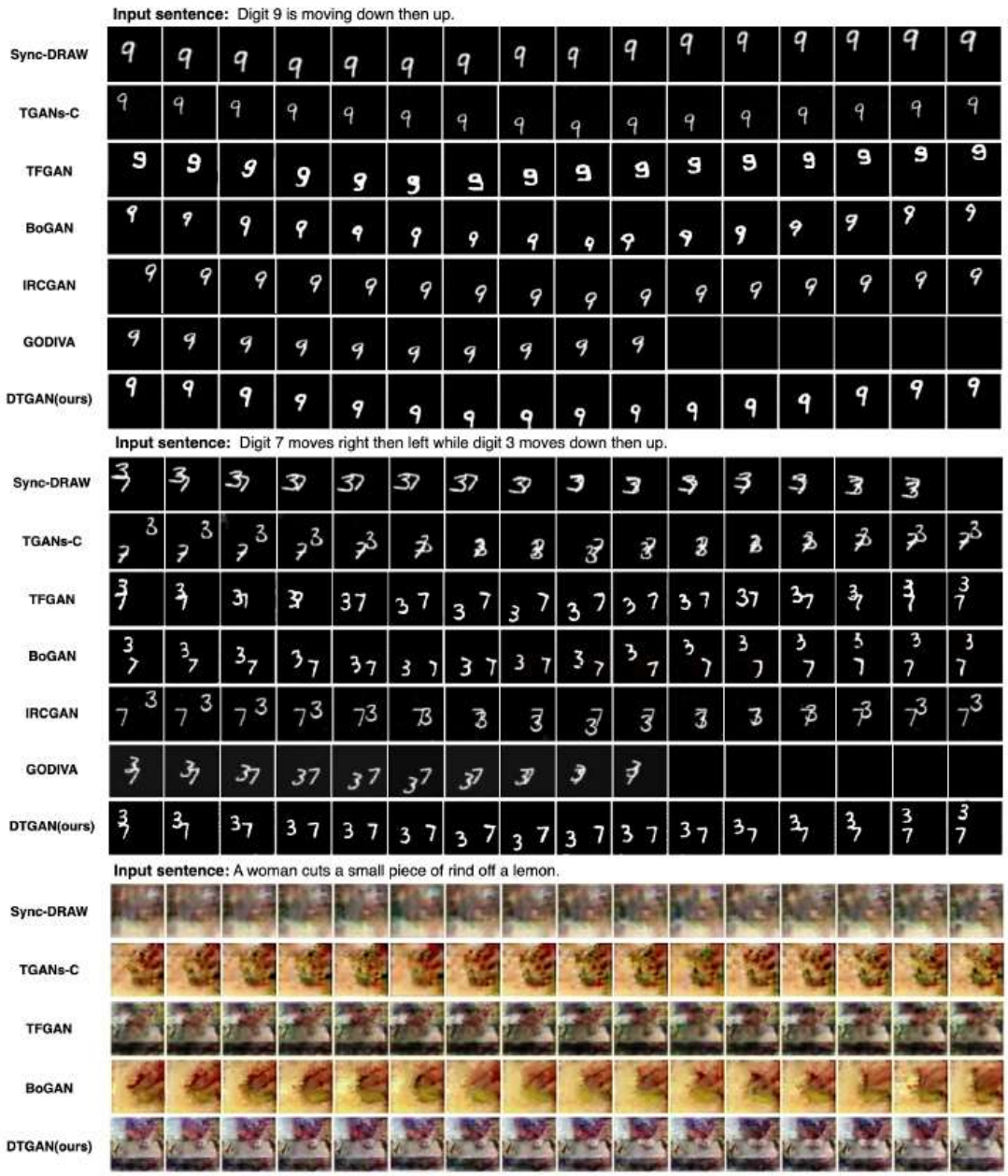

Figure 5. Qualitative comparison on SBMG, TBMG, and MSVD.

Table 3. Quantitative results of ablation study for different components of the proposed method, evaluated on SBMG.

Method	FID	FID2vid
Baseline	44.21	3.05
With LKD	**27.52**	3.19
With VRU	41.11	2.78
LKD+VRU (ours)	28.78	**2.73**

The videos generated by each ablation study are shown in **Figure 6**. 1) With the constraints of KD loss, all frames contain digit 2 with good shapes, while the digit still has slight motions in up and down directions. 2) We can also see that frames generated by the model with VRU can better reflect the motion of "right then left".

But several frames containslight distortion. Finally, by using all the proposed modules, we combine the advantages of 1) and 2) and eliminate their disadvantages, thus meeting our expectations: semantically consistent across text-visual modalities and temporally coherent across video frames.

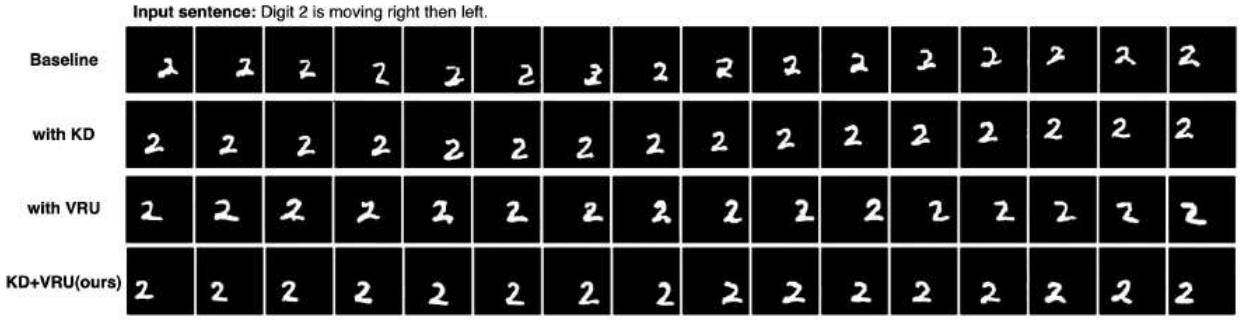

Figure 6. Qualitative results of ablation study for different components of the proposed method, evaluated on MSVD.

F. User study:

Current existing automated evaluation metrics are useful but imperfect. So we conduct a user study to further evaluate our model. We invite 30 users to evaluate the text-visual semantic consistency and temporal coherence of the results (15 inquiries each part). We choose four different methods for comparison, namely Sync-DRAW [35], TGANs-c [9], TFGAN [4], and BoGAN [10].

All these methods are trained and tested on the TBMG dataset. We randomly select 75 videos from testing results in total, 15 for each method.

For the semantic consistency evaluation, we randomly select six frames as the sample representations from each of the 75 video samples. In each inquiry, users will see five sample representations from the five methods, which are conditioned on the same input text. We ask the users to score the sample representations from 1 to 5, and a higher score means that the digits in the sample representations are more realistic and more consistent with the input text. We count the times of each method getting each score and show it in **Figure 7a**. For temporal coherence evaluation, users will see a sentence and its corresponding five GIFs in each inquiry. Scoring rules and the data collecting methods are the same as the semantic evaluation. Note that higher scores here indicate that the GIF image is more coherent. The results are shown in **Figure 7b**. From **Figure 7**, we can find that users think our generated videos are more coherent and more consistent with the given text than videos generated by the other four methods.

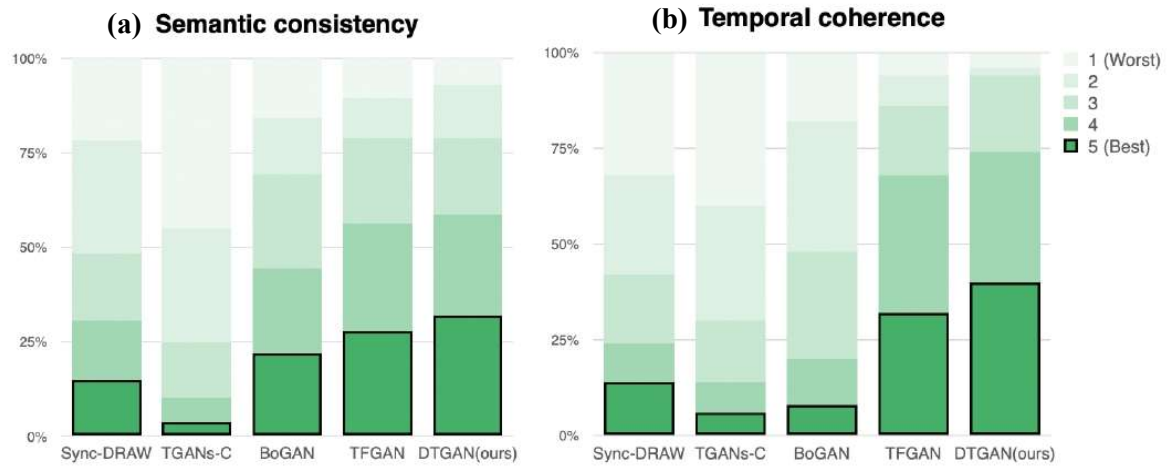

Figure 7. User study on TBMG dataset. User scores (anatomized and order randomized) from 5 (best) to 1.

5. Conclusion

In this paper, we have proposed a Distillation and Translation GAN (DTGAN) to generate videos from text. First, we use KD to promote text-visual semantic consistency through transferring intermediate layer information and perceptual information from a T2I teacher model. Second, we specially design a visual recurrent unit (VRU) for frame sequences generation. With VRU, our generator imitates the NMT models to synthesize frames iteratively and resolve temporal coherence. Our method has some limitations. It is difficult to align semantically with longer complex texts, which is a research direction for our future work.

Author contributions: Algorithm ideas, HH and ZL; experiments and writing, HH; guiding suggestions, FF and ZL; writing—original draft preparation, HH; writing—review and editing, FL and CX. All authors have read and agreed to the published version of the manuscript.

Conflict of interest: The authors declare no conflict of interest.

References

1. Dong X, Long C, Xu W, et al. Dual Graph Convolutional Networks with Transformer and Curriculum Learning for Image Captioning. Proceedings of the 29th ACM International Conference on Multimedia. Published online October 17, 2021. doi: 10.1145/3474085.3475439
2. Fang F, Yi M, Feng H, et al. Narrative Collage of Image Collections by Scene Graph Recombination. IEEE Transactions on Visualization and Computer Graphics. 2018; 24(9): 2559-2572. doi: 10.1109/tvcg.2017.2759265
3. Fang F, Luo F, Zhang HP, et al. A Comprehensive Pipeline for Complex Text-to-Image Synthesis. Journal of Computer Science and Technology. 2020; 35(3): 522-537. doi: 10.1007/s11390-020-0305-9
4. Balaji Y, Min MR, Bai B, et al. Conditional GAN with Discriminative Filter Generation for Text-to-Video Synthesis. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. Published online August 2019. doi: 10.24963/ijcai.2019/276
5. Deng K, Fei T, Huang X, et al. IRC-GAN: Introspective Recurrent Convolutional GAN for Text-to-video Generation. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. Published online August 2019. doi: 10.24963/ijcai.2019/307
6. Zhang H, Xu T, Li H, et al. StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2019; 41(8): 1947-1962. doi: 10.1109/tpami.2018.2856256

7. Xu T, Zhang P, Huang Q, et al. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Published online June 2018. doi: 10.1109/cvpr.2018.00143
8. Zhang Z, Xie Y, Yang L. Photographic Text-to-Image Synthesis with a Hierarchically-Nested Adversarial Network. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Published online June 2018. doi: 10.1109/cvpr.2018.00649
9. Pan Y, Qiu Z, Yao T, et al. To Create What You Tell. Proceedings of the 25th ACM international conference on Multimedia. Published online October 23, 2017. doi: 10.1145/3123266.3127905
10. Chen Q, Wu Q, Chen J, et al. Scripted Video Generation With a Bottom-Up Generative Adversarial Network. IEEE Transactions on Image Processing. 2020; 29: 7454-7467. doi: 10.1109/tip.2020.3003227
11. Mazaheri A, Shah M. Video Generation from Text Employing Latent Path Construction for Temporal Modeling. 2022 26th International Conference on Pattern Recognition (ICPR). Published online August 21, 2022. doi: 10.1109/icpr56361.2022.9956706
12. Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. Advances in neural information processing systems. 2014.
13. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv. 2014; arXiv:1409.0473.
14. Reed S, Akata Z, Yan X, et al. Generative adversarial text to image synthesis. International Conference on Machine Learning. PMLR 2016.
15. Kim D, Joo D, Kim J. TiVGAN: Text to Image to Video Generation With Step-by-Step Evolutionary Generator. IEEE Access. 2020; 8: 153113-153122. doi: 10.1109/access.2020.3017881
16. Li Y, Min M, Shen D, et al. Video Generation From Text. Proceedings of the AAAI Conference on Artificial Intelligence. 2018; 32(1). doi: 10.1609/aaai.v32i1.12233
17. Yamamoto S, Tejero-de-Pablos A, Ushiku Y, et al. Conditional video generation using actionappearance captions. arXiv preprint arXiv:1812.01261 (2018).
18. Tulyakov S, Liu MY, Yang X, et al. MoCoGAN: Decomposing Motion and Content for Video Generation. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Published online June 2018. doi: 10.1109/cvpr.2018.00165
19. Liu Y, Wang X, Yuan Y, et al. Cross-Modal Dual Learning for Sentence-to-Video Generation. Proceedings of the 27th ACM International Conference on Multimedia. Published online October 15, 2019. doi: 10.1145/3343031.3350986
20. Gupta T, Schwenk D, Farhadi A, et al. Imagine this! scripts to compositions to videos. Proceedings of the European conference on computer vision (ECCV). 2018.
21. Wu C, Huang L, Zhang Q, et al. Godiva: Generating open-domain videos from natural descriptions. arXiv preprint arXiv:2104.14806 (2021).
22. Wu C. "N\" uwa: visual synthesis pre-training for neural visual world creation. arXiv preprint arXiv:2111.12417 (2021).
23. Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015).
24. Aguinaldo A, Chiang P Y, Gain A, et al. Compressing gans using knowledge distillation. arXiv preprint arXiv:1902.00159 (2019).
25. Johnson J, Alahi A, Li FF. Perceptual losses for real-time style transfer and super-resolution. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam; 11–14 October, 2016; The Netherlands. 2016.
26. Chen H, Wang Y, Shu H, et al. Distilling Portable Generative Adversarial Networks for Image Translation. Proceedings of the AAAI Conference on Artificial Intelligence. 2020; 34(04): 3585-3592. doi: 10.1609/aaai.v34i04.5765
27. Li M, Lin J, Ding Y, et al. GAN Compression: Efficient Architectures for Interactive Conditional GANs. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Published online June 2020. doi: 10.1109/cvpr42600.2020.00533
28. Jin Q, Ren J, Woodford OJ, et al. Teachers Do More Than Teach: Compressing Image-to-Image Models. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Published online June 2021. doi: 10.1109/cvpr46437.2021.01339
29. Li S, Lin M, Wang Y, et al. Learning Efficient GANs for Image Translation via Differentiable Masks and Co-Attention Distillation. IEEE Transactions on Multimedia. 2023; 25: 3180-3189. doi: 10.1109/tmm.2022.3156699
30. Cortes C, Mohri M, Rostamizadeh A. Algorithms for learning kernels based on centered alignment. The Journal of Machine

Learning Research. 2012; 13(1): 795-828.
31. Cristianini N, Shawe-Taylor J, Elisseeff A, et al. On Kernel-Target Alignment. Advances in Neural Information Processing Systems 14. Published online November 8, 2002: 367-374. doi: 10.7551/mitpress/1120.003.0052
32. Kornblith S, Norouzi M, Lee H, et al. Similarity of neural network representations revisited. International Conference on Machine Learning. PMLR, 2019.
33. Zagoruyko S, Komodakis N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. arXiv preprint arXiv:1612.03928 (2016).
34. Heusel M, Ramsauer H, Unterthiner T, et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in Neural Information Processing Systems 30 (2017).
35. Mittal G, Marwah T, Balasubramanian VN. Sync-DRAW. Proceedings of the 25th ACM international conference on Multimedia. Published online October 19, 2017. doi: 10.1145/3123266.3123309