Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.041219

ARTICLE

# Automated Video Generation of Moving Digits from Text Using Deep Deconvolutional Generative Adversarial Network

**Anwar Ullah[1], Xinguo Yu[1,*] and Muhammad Numan[2]**

[1]National Engineering Research Center for E-Learning, Central China Normal University, Wuhan, 430079, China

[2]Wollongong Joint Institute, Central China Normal University, Wuhan, 430079, China

*Corresponding Author: Xinguo Yu. Email: xgyu@gmail.ccnu.edu.cn

Received: 14 April 2023 Accepted: 19 June 2023 Published: 29 November 2023

## ABSTRACT

Generating realistic and synthetic video from text is a highly challenging task due to the multitude of issues involved, including digit deformation, noise interference between frames, blurred output, and the need for temporal coherence across frames. In this paper, we propose a novel approach for generating coherent videos of moving digits from textual input using a Deep Deconvolutional Generative Adversarial Network (DD-GAN). The DD-GAN comprises a Deep Deconvolutional Neural Network (DDNN) as a Generator (G) and a modified Deep Convolutional Neural Network (DCNN) as a Discriminator (D) to ensure temporal coherence between adjacent frames. The proposed research involves several steps. First, the input text is fed into a Long Short Term Memory (LSTM) based text encoder and then smoothed using Conditioning Augmentation (CA) techniques to enhance the effectiveness of the Generator (G). Next, using a DDNN to generate video frames by incorporating enhanced text and random noise and modifying a DCNN to act as a Discriminator (D), effectively distinguishing between generated and real videos. This research evaluates the quality of the generated videos using standard metrics like Inception Score (IS), Fréchet Inception Distance (FID), Fréchet Inception Distance for video (FID2vid), and Generative Adversarial Metric (GAM), along with a human study based on realism, coherence, and relevance. By conducting experiments on Single-Digit Bouncing MNIST GIFs (SBMG), Two-Digit Bouncing MNIST GIFs (TBMG), and a custom dataset of essential mathematics videos with related text, this research demonstrates significant improvements in both metrics and human study results, confirming the effectiveness of DD-GAN. This research also took the exciting challenge of generating preschool math videos from text, handling complex structures, digits, and symbols, and achieving successful results. The proposed research demonstrates promising results for generating coherent videos from textual input.

## KEYWORDS

Generative Adversarial Network (GAN); deconvolutional neural network; convolutional neural network; Inception Score (IS); temporal coherence; Fréchet Inception Distance (FID); Generative Adversarial Metric (GAM)

## 1 Introduction

In the currently advanced era, Artificial Intelligence (AI) is an active field with many significant applications and valuable research topics. AI is transforming every area of life and is a tool that

allows people to rethink how the data is analyzed, integrate the information, and use the resulting insights to improve decision-making. Among the various AI techniques, machine and deep learning have gained widespread attention from researchers due to their ability to power numerous applications such as image classification, multimedia concept retrieval, text mining, video recommendations, and much more [1]. In deep learning, the layered concept is used to represent data abstraction to build computational models, and algorithms such as convolutional neural networks and generative adversarial networks have completely changed the perception of information processing. Therefore, deep learning has prevailed in the field of artificial intelligence.

Generative Adversarial Networks (GANs) [2], proposed by Goodfellow et al. in 2014, are one of the deep learning models based on zero-sum game theory, where the total gains of two players are zero, and the gain or loss of each player's utility is precisely balanced [3]. GANs often simultaneously involve a Generator (G) and a Discriminator (D) learning. The G attempts to capture the potential distribution of the real samples and creates new data samples. At the same time, the D is often a binary classifier to distinguish the real samples from the generated samples as accurately as possible. Thus, G and D inherited the structure of the currently popular deep neural networks [4,5]. The GAN optimization process is a minimax game process, and the goal is to achieve Nash equilibrium [6], which assumes that the G has captured the distribution of real samples. Therefore, the importance of this emerging generative model is to preserve data distribution through unsupervised learning and generate more realistic/actual data [3]. GANs have been extensively studied due to their massive application viewpoint, including language, image, video processing, etc.

Intelligent graphic design tools have the potential to generate engaging and informative videos that help people learn about the world around them. However, these tools can be challenging and inaccessible, especially for those with limited technical knowledge and resources. Therefore, an intelligent system capable of performing text-based video editing tasks is necessary to make video creation easier for people with less extensive technical expertise. Such techniques can be applied across various domains, including gaming, virtual reality, and educational materials. In Computer Vision (CV), the use of Generative Adversarial Networks (GANs) for the automatic generation of visual content is a significant advancement, enabling the creation of highly realistic images and videos. By incorporating GANs into intelligent video editing systems, accessibility and ease of use can be improved, empowering more people to create compelling visual content. Significant research has been devoted to improving the quality of results in various fields, including these GAN for super-resolution images [7,8], for image and video classification [9,10], respectively, for cartoon images generation from face photos [11], for Computed Tomography (CT) image denoising [12], etc. GANs are also particularly effective in tackling complex tasks such as multi-domain synthesis [13,14], and multi-view generation [15]. These techniques have demonstrated remarkable success in image denoising, high-resolution, and generating high-quality images and videos, making them valuable tools for various applications.

Video generation from text is a complex task compared to image generation [16–19] because videos are a complex sequence of individual images that follow spatial and temporal dependencies and are more difficult due to semantic alignment requirements between text and video at frame and video levels [20]. Realistic and synthetic video generation from text is a difficult task because there are multiple issues, such as capturing the coherence between individual frames, digit deforming, noises between co-related structures, blur output, and temporal coherence between frames. For example, a hybrid Variational Auto Encoder and Generative Adversarial Network (VAE-GAN)-based model was proposed by Li et al. [21] to first produce a "gist" of the video from the given text using VAE, where the "gist" is an image that specifies the background color and object arrangement. Then, based on

the gist, the video's motion and substance are created using GAN. Unfortunately, the movements of generated video are incoherent since they ignore the relationship between successive frames.

On the other hand, Pan et al. [22] paid attention to the temporal coherence between frames and produce video using the input text using a properly conceived discriminator. However, because text and video are aligned broadly based on the classical conditional loss [16], some precise semantic words essential for synthesizing the details are ignored in the resulting videos. Moreover, Chen et al. [20] proposed a Bottom-Up Generative Adversarial Network (BoGAN) model to deal with multi-level alignment and coherence problems using region-level, frame-level, and video-level losses; the results are competitive. However, they are facing some overshoot because of multiple levels and 3D deconvolution. In contrast to 1D and 2D deconvolution, 3D deconvolution results in a more significant loss of information. While applying the video-level discriminator to the whole video, they need to pay more attention to the significant words for a better result, causing some incoherent problems in the adjacent frames.

This research proposes a novel solution for generating moving digit videos from text using the Deep Deconvolutional Generative Adversarial Network (DD-GAN). This research focuses on generating preschool math videos from a text that exhibits spatial and temporal coherence. Using DD-GAN, we can generate synthetic videos for addition, subtraction, multiplication, and division from written text, potentially significantly improving children's math education. Some of the generated videos of DDGAN can be seen in Fig. 1 using SBMG, TBMG, and custom math video datasets.

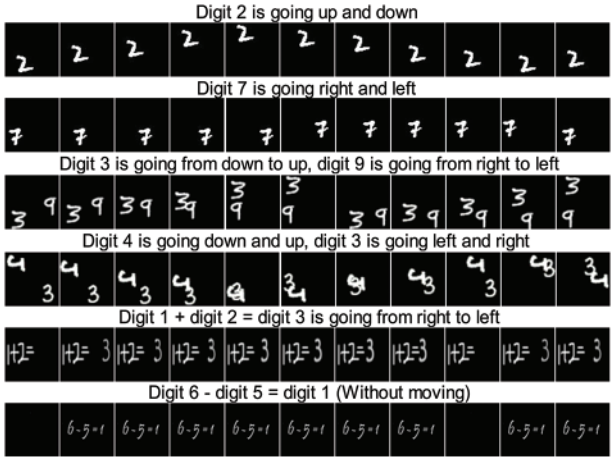


**Figure 1:** The experimental results of our DD-GAN on the SBMG, TBMG, and custom math videos datasets

In the DD-GAN architecture, we start with an LSTM-based text encoder that transforms the text description into a text embedding ($\varphi_t$) and then applies a conditioning augmentation (CA) technique [18] to the embedding, which helps to smooth the text in multiple ways. The resulting text embedding $\varphi_t$ is then combined with a conditioning extension variable ć and a random noise variable $z$ and fed through a deep deconvolutional neural network (DDNN) that serves as the Generator (G) to produce the video. The DDNN model uses an inverse convolution approach, which allows it to generate videos with spatial and temporal coherence that match the input text description. A modified deep convolutional neural network (DCNN) serves as the Discriminator (D), which is used to distinguish between the generated and real videos. The DDNN and DCNN networks are trained using an adversarial training scheme. The DDNN network is trained to fool the DCNN network

into thinking that the generated frames are real, while the DCNN network is trained to classify the generated frames as fake correctly. This feedback loop between the DDNN and DCNN networks allows the DDNN to improve over time and generate more realistic videos. The DCNN model has been modified to enhance its ability to identify differences between the generated and real videos, as described later in this paper. First, we trained our DD-GAN with two synthetic datasets called Single Digit Bouncing Mnist-4 (SBMG) [22] and Two Digit Bouncing Mnist-4 (TBMG) [22]. It then trained on a custom early school mathematics video dataset (Math Dataset). For the quantitative assessment, we used the most important and commonly used metrics, namely: (1) Inception Score (IS) [23], (2) Fréchet Inception Distance (FID) [24], (3) Fréchet Inception Distance 2 Video (FID2vid) [25], and (4) Generative Adversarial Metric (GAM) [26]. In addition, we also conducted a human study for further evaluation based on the realism, relevance, and coherence of the generated videos. As a result, our DDGAN performs better than other state-of-the-art methods. Fig. 2 showcases the simplified architecture of DD-GAN, and Fig. 3 shows the simplified architecture of other state-of-the-art methods. In contrast, the complete architecture of the DD-GAN can be seen later in this paper.

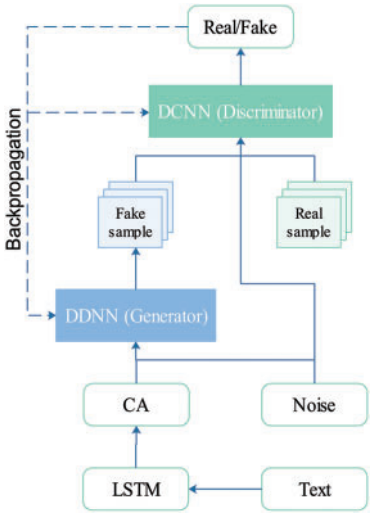


**Figure 2:** Simple architecture of our proposed DD-GAN model

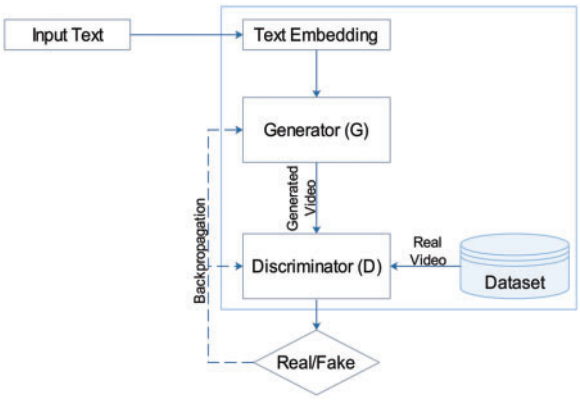


**Figure 3:** Simple architecture of the other state-of-the-art text-to-video GAN model

The contribution to the proposed DD-GAN is summarized as follows:

- Firstly, we introduced a novel Deep Deconvolutional Generative Adversarial Network (DD-GAN) that generates high-quality moving digit videos from written text. The generated frames are closely temporal coherent with the given scripts.
- Secondly, we used a Deep Deconvolutional Neural Network (DDNN) as a Generator (G) to generate moving digits and semantically matched video from the input text due to its deep architecture and proposed several modifications to the Deep Convolutional Neural Network (DCNN) as a Discriminator (D) to utilize the conditional information provided by the input text and effectively distinguish between generated and real videos.
- Thirdly, we tested the performance of the DD-GAN model on Mnist SBMG and TBMG datasets, as well as a custom-generated mathematics video dataset. We evaluated the results using standard metrics such as Inception Score (IS), Fréchet Inception Distance (FID), FID2vid, and Generative Adversarial Metric (GAM).
- Fourthly, we also conducted a human study to assess realism, relevance, and coherence and demonstrated significant improvements in metrics and human analysis, thus proving the effectiveness of the proposed approach.

This research work is structured as follows: Section 2 provides a comprehensive overview of related work, examining various video-generated GAN models. The proposed methodology is then detailed in Section 3 before presenting the results of our experiments and a thorough discussion of our findings, including limitations, in Section 4. Finally, we conclude our research in the last section.

## 2 Related Works

The increasing trend of AI Generative Adversarial Networks (GANs) models and algorithms for automatic content creation in various media, entertainment, and education sectors has sparked an increasing interest in automatically generating content such as text, audio, images, and videos. Different variations of the GAN models are covered in various reviews and research papers for more details about the generation and synthesis of multimedia content (images, videos, and audio), along with some applications, classification, challenges, and performance of various GAN models presented by Lalit and Singh [27]. But this related work focuses only on the video GANs models, which is the nature of our research topic. Vondrick et al. [28] made the first attempt in 2016 in this direction to produce video sequences. This GAN uses a two-stream technique, with one stream concentrating on the static background and the other on the foreground. They proposed a GAN architecture based on 3D convolutions; they generated encouraging results; closer observation showed that it could only produce fixed-length, somewhat noisy, and lacking in object structural integrity videos. In contrast to 1D and 2D deconvolution, 3D deconvolution results in a more significant loss of information. The comprehensive review of the video GAN models discussed briefly is based on two main divisions of condition, with our focus being on the conditional category of text-to-video GANs.

### *2.1 Unconditional Video Generation*

The unconditional video GANs are those with an unsupervised frame where the videos are produced without prior information [29]. The model must capture the data distribution without an input signal wizard that can help narrow the target gap. Training unconditional video GANs is so complicated that some of them become the foundation for conditional frameworks, like Motion Content GAN (MoCoGAN) [30] is an unconditional video GANs model used in conditional video GANs models such as storyGAN [31], and Text Filter Conditioning GAN (TFGAN) [32].

### *2.2 Conditional Video Generation*

Several works used conditional signals in GANs to control the modes of generated data. These conditions may be based on images [33–35], semantic maps [36–39], audio signals (speech) [40–43], or video [44–47], but due to the nature of our DD-GANs model, we reviewed and explained the current work based on textual conditions in Section 2.3.

### *2.3 Text-to-Video Generation*

Text-to-Video GAN's models focus on two main purposes: producing video according to conditional text. Firstly, to maintain semantically aligned consistency between the given text condition and video generation, and secondly, to generate realistic videos to maintain consistency and coherence in the frames. Mittal et al. [48] developed a method that captures the time-dependent sequence of frames to merge a variational autoencoder with a recurrent attention mechanism. This method was the first text-to-video generation approach implemented in 2017. They addressed some of the drawbacks in [28], especially the lack of object structural integrity in videos, and they upheld the objects' structure to a significant extent. An improved model [49] was later proposed, namely (Cap2vid), in which the short- and long-term dependencies between frames and generated videos are incrementally integrated. They specially addressed the spatiotemporal semantics of the video, thus generating good-quality videos from the caption.

Onward, in 2018, Pan et al. [22] proposed a novel Temporal Generative Adversarial Network Conditioning on Captions (TGANs-C), where the input of the Generator (G) was random noise along with a caption. Where the latent noise vector and caption embedding are combined as the input to the generator network, which is then used to create a frame sequence using 3D spatiotemporal convolutions, they tried to overcome the temporal and semantic coherence-dependent frame problems and successfully overcome them somehow. In contrast, the GANs model proposed in [21] generated videos using a two-step VAE based on input text to generate a gist of the video. The gist, which is an image, gives a background color and layout of the object, and after that, the video content and motion are generated. The authors tried to address the mode-collapse problems in the frames. In some results, they achieved competitive results.

However, reference [32] introduced a TFGAN method with multiscale text conditions to generate convolution filters that extract the text features from the coded text. After generating the convolution filter, they are fed into the discriminator to facilitate and strengthen the links between text and video and generate some competitive samples. Contrary to this, the story [31] method, a story visualization model based on the condition of multiple sentences, contains a context encoder and a story encoder. Unlike other video GAN models, storyGAN focuses less on the stability of motion and instead on the global consistency of the story, resulting in videos with a coherent storyline rather than just smooth motion.

The Cap2vid [49] model was improved by incrementally integrating short- and long-term dependencies between frames and generated videos, addressing spatiotemporal semantics, and generating good-quality videos from the caption. TGANs-C [22] proposed a novel approach to overcome temporal and semantic coherence-dependent frame problems by using 3D spatiotemporal convolutions to create a frame sequence from a combination of latent noise vectors and caption embedding as input to the generator network. The GANs model proposed in [21] attempted to address mode-collapse problems in the frames by generating videos using a two-step VAE based on input text to generate a GIST of the video, followed by video content and motion. TFGAN [32], with multiscale text conditions, introduced a method called TFGAN that generates convolution filters that extract the

text features from the coded text and feeds them into the discriminator to facilitate and strengthen the links between text and video. StoryGAN [31], a story visualization model based on the condition of multiple sentences, focuses less on the stability of motion and instead on the global consistency of the story.

While all the methods discussed above produce positive results but still have limitations that need to be addressed in future studies, the more profound studies demonstrated that their generations need more abjectness and are generally noisy, incoherent, low-quality, and fixed in duration. The ability of video generation models to produce high-quality, coherent, and realistic videos that are an exact match for the input text description falls short of the current state-of-the-art. Video generation models aim to teach machines how to generate videos from textual descriptions. This involves advanced techniques in natural language processing, computer vision, and machine learning. These models have many potential applications, such as video summarization, content creation, and virtual reality. By improving these models, machines can better understand human language and generate useful videos for various tasks. Considering the coherence issue, DD-GANs combine video-text semantic matching and frame coherence to generate realistic, coherent videos that match the given text description. This GAN-based approach ensures synthetic videos are visually convincing while maintaining fidelity to the text. It represents a promising direction for advancing video synthesis and semantic matching.

## 3 Proposed Methodology

The fundamental difficulties in generating video from text lie in capturing both spatial and temporal coherence and the semantic relationship between text and video. This research addresses the temporal coherence problem and proposes a novel approach called Deep Deconvolutional Generative Adversarial Network (DD-GAN) for generating moving digit videos from text descriptions. DD-GAN is designed to overcome the challenges of text-to-video generation, particularly in generating synthetic and early school mathematics videos from the text.

GAN is a deep neural network consisting of a Generator (G) and a Discriminator (D). These two components are trained competitively, where G generates new data while D authenticates the data. In our proposed method, the Generator (G) is a deep deconvolutional neural network that generates the data. At the same time, the Discriminator (D) is a modified deep convolutional neural network that authenticates the data.

The overall diagram of our proposed DD-GAN method is shown in Fig. 4, which includes the Long Short Term Memory (LSTM)-based text-encoder, Conditional Augmentation (CA), DDNN as a Generator (G), and DCNN as a Discriminator (D) with every step and process. In the following sections, we will present the details of our novel proposed method.

### *3.1 Text Encoder Network*

To convert the provided text description into $z_{\text{text}}$ latent code vectors or machine-readable codes for video generation, we used an LSTM-based encoder. Words from a phrase are fed one by one in sequence to the encoder at each time step during the testing and training phases. For example, if the text "Digit 2 is moving up and down" is used, the word "Digit" is fed at time step $t = 1$, the word 2 is provided at time step $t = 2$, and so on. Each word is first represented as a single-hot vector. As a result, $\{w_1, w_2, \ldots, w_n\}$ can be used to express a sentence of length n. The single-hot vector for the *t-th* word is $w_t$. After that, the sentence is sent into a bidirectional LSTM network, which contextualizes each word in $h_t$. Next, we use an LSTM-based encoder to input the contextually embedded word sequence $\{h_1, h_2, \ldots, h_n\}$, and use the latent text code $z_{\text{text}} \in \mathbb{R}^{d\text{text}}$ as the final LSTM output. After getting the

embedding output, Conditioning Augmentation (CA) techniques are used to further extract valuable features.

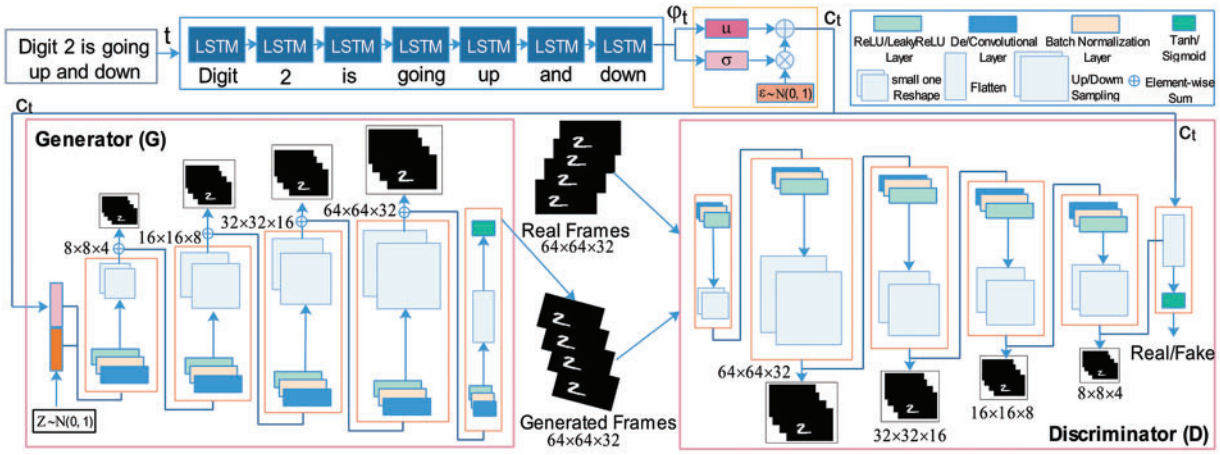


**Figure 4:** The framework of our proposed DD-GAN text-to-video generation model. A first LSTM-based encoder is used for text-embedding, Conditioning Augmentation (CA) for smooth condition manifolds, Generator (G) for generating the data, and Discriminator (D) for authenticating the data

### *3.2 Conditional Augmentation (CA) Technique*

The majority of text-to-video models obtain the text and encode it using an encoder, resulting in text-embedding ($\varphi_t$). A text-embedding ($\varphi_t$) is a nonlinear transformation that produces a latent dependent variable for an independent generator variable (G). Furthermore, the latent space for the $\varphi_t$ is extremely high-dimensional, exceeding 100 dimensions. As a result, if there are fewer data or small datasets, the latent data manifold is more likely to break or discontinue, which is undesirable for the generator (G). This issue is vast because it causes blurry artefacts, incoherence, and frame disconnectivity. To mitigate the problem, Zhang et al. [18] proposed a novel technique called Conditioning Augmentation (CA), which adds a new conditioning variable `c to the equation. Due to the inconsistency with the fixed or static conditioned text variable "c," the latent variable `c was computed using an independent or autonomous Gaussian Distribution $\boldsymbol{N}$ ($\mu(\varphi_t)$, $\sum(\varphi_t)$), where $\mu(\varphi_t)$, stands for the mean (average) and $\sum(\varphi_t)$ stands for principle diagonal covariance matrix. The proposed conditioning augmentation provides better outcomes and more training pairs, resulting in the smallest number of text and frames pairs possible, motivating robustness to share fewer changes with the conditioning phase. To perfect the conditioning process and avoid overfitting.

While training, we introduced a specific condition to enhance the performance of G. This condition involved the equation concerning divergence, named the Kullback–Leibler (KL) divergence, between the standard Gaussian distribution and the conditioning Gaussian distribution. This innovative approach aims to achieve smoothness in the condition manifold. Additionally, we employed CA to improve the performance of the proposed method and the conditional manifold's smoothness and robustness.

$$D_{K\mathcal{L}}(N\left(\mu\left(\varphi_t\right), \sum\left(\varphi_t\right)\right) \; ||N\left(0,1\right)) \tag{1}$$

### *3.3 Deep Deconvolutional Neural Network as Generator (G)*

Generating videos from semantic texts faces two main challenges: first, extracting relevant information from sentences and linking it to video content, which can be addressed with advances in Natural Language Processing (NLP) and multimodal learning. Second, accurately modeling both long- and short-term temporal dependencies of video frames, including slow-changing global and fast-changing local patterns. Existing techniques focus only on fast-changing patterns with 3D convolutions or treat each frame generation independently, ignoring temporal dependency modeling.

In this research, we proposed a novel approach for generating moving digit videos from text, specifically targeting synthetic and preschool mathematics scripted video generation. The proposed method, Deep Deconvolutional Generative Adversarial Network (DD-GAN), effectively captures the long-term temporal dependencies between frames and ensures coherence between individual frames during video generation. Through advanced deep learning techniques and a carefully crafted network architecture, our method produces high-quality videos that accurately reflect the intended content of the input text. Fig. 4 illustrates our approach and demonstrates its effectiveness in generating realistic and coherent videos. The DD-GAN has a deep spatiotemporal architecture, whose input combines embedded text $\varphi_t$ and an extra variable of Conditioning Augmentation (CA) `c and noise $z$. The result is a sequence of generated video frames. In more detail, each word in the bi-directional LSTM relates to 2 hidden states, one for the forward direction and one for the backward direction. As a result, we combine its 2 hidden states to express a word's semantic meaning. $E \in \mathbb{R}^{T \times w_n}$ represents the feature matrix for all words. The ith column represents the embedding of the ith word, $e_i$. T represents the dimension of the extracted word, while $W_n$ represents the number of words in a sentence. The actual sentence representation, $\grave{e} \in \mathbb{R}^T$, concatenates the bidirectional LSTM's latest hidden states. We combine the semantic text embedding `e with the additional variable of Conditioning Augmentation (CA) `c and random noise $z \in \mathbb{R}^{T_z} \sim N(0, 1)$ to generate a video with the same semantic content as the input text. Then we learn a unified embedding using a fully-connected layer.

$$P = \mathcal{W}_e [z + \grave{c} ; \grave{e}] \in \mathbb{R}^{t_p \times (t_z + T)} \tag{2}$$

where $\mathcal{W}_e \in \mathbb{R}^{t_p \times (t_z + T)}$ represents the embedding weight, and $t_p$ is used to represent the embedding size; moreover, (;) represents the concatenation operation. Below is the actual input for the Generator (G) to produce the video.

$$V_n = G\ (D_n),\ n = 1, \ldots,\ l \tag{3}$$

Here, $n = 1, \ldots, l$, and $l$ represent the number of frames in the video. In $G$, weight sharing is commonly employed in both the temporal and spatial directions to reduce the number of parameters and improve model stability. The entire network can be viewed as a nonlinear mapping between semantic texts and desired videos. For a frame-wise generation, there is a deep deconvolution neural network in the spatial direction at each time step. Deconvolution, batch-normalization, and ReLU activation layers are employed first, followed by another deconvolution and up-pooling (Up-sample) layer to upscale the frame size by a factor of two across consecutive levels of feature maps. This process is repeated several ($n = 5$) times to generate a 64 $\times$ 64 size video containing 32 frames. Every step is a memory unit, so the state of these memory units is communicated between frames, allowing each frame to be built based on historical data, contributing to temporal coherence. Finally, we combine all of the created frames into a single video with a size of $t_l \times t_c \times t_h \times t_w$, or we can represent the whole video as:

$$\chi_{video} = G\ (p) \in \mathbb{R}^{t_l \times t_c \times t_h \times t_w} \tag{4}$$

Here, $\chi_{video}$ represents the generated video, and $G\ (p)\ \epsilon\ \mathbb{R}^{t_l \times t_c \times t_h \times t_w}$ denotes for the $\mathrm{i}^{\mathrm{th}}$ generated frame. Where $t_l$ is the channel number, $t_c$ is the sequence length, and $t_h$ and $t_w$ is used for frame height and width. Due to the deep deconvolutional scheme, our DD-GAN can generate synthetic, plausible, and preschool mathematics videos from text with realism, relevance, and coherence between frames. The cost function that DD-GAN uses is shown in the equation below:

$$\mathrm{S_K}\ (\Theta) = \sum G\,(y|\vartheta)\,.\ Q\,(y.\vartheta)\ldots \tag{5}$$

where is the length of the text, and $G$ is the Generator function for video generation. $\vartheta$ is the generator function's stochastic gradient that will be optimized by the generator. Moreover, we utilize our work by using the Wasserstein GAN [50] formulation, which is given below:

$$\underset{G}{min}\ \underset{D}{max}\ \mathbb{E}_{V\sim p(v)}\left[D\,(\mathcal{V};\varnothing_d)\right] - \mathbb{E}_{z_v\sim p(z_v)}\left[D\left(G\left(z_v;\varnothing_g\right);\varnothing_d\right)\right] \tag{6}$$

Here, $D$ represents the discriminator, $G$ represents the generator, $\varnothing_d$ and $\varnothing_g$ are used for maintaining maximum and minimum Lipschitz constants of the function, and z represents random noise. Other equations used in our proposed model are given below:

For Generator (G) and Discriminator (D) losses, we used the following equations, respectively:

$$\nabla\varnothing_g \frac{1}{i}\sum\nolimits_{n=1}^{i} log\,(1 - \mathrm{D}\,(\mathrm{G}_z\,(\mathrm{n}))) \tag{7}$$

$$\nabla\varnothing_d \frac{1}{i}\sum\nolimits_{n=1}^{i} log\,(\mathrm{D}\,(\mathrm{n}) + \log\,(1 - D\,(G_z\,(\mathrm{n})))) \tag{8}$$

For Generator (G) accuracy, we used the below equation:

$$GA = COR = \frac{RS}{A} \tag{9}$$

where $GA$ means generator accuracy, $COR$ is the currently organized rate, $RS$ represents the correctly recognized samples, and $A$ denotes the number of all samples.

And for Discriminator (D) accuracy, we used the below equation:

$$DA = COR = \frac{AS - ES}{AS} \tag{10}$$

where $DA$ means discriminator accuracy, similarly, $COR$ denotes the currently organized rate, $AS$ represents the accepted samples, and finally, $ES$ is the experimental sample.

### *3.4 Modified Deep Convolutional Neural Network as Discriminator (D)*

The discriminator's role is crucial in determining the authenticity of a video, as it categorizes it into two distinct categories: "genuine" and "fake". To differentiate between actual and generated videos, the discriminator community relies on three perspectives: (1) the complete video, (2) every video frame, and (3) the movement throughout adjacent frames. Deep Convolutional Neural Network (DCNN) has proven to be an effective and accurate tool for classifying imaging and video data. Therefore, we utilized modified DCNN as a discriminator to distinguish between genuine and fake videos.

$$D\,(m_v)\ : \mathbb{R}^{t_l \times\ t_c \times t_h \times t_w} \rightarrow [0,\ 1] \tag{11}$$

$D$ extracts $m_v$ video-level features from the input video first $V \in \mathbb{R}^{t_l \times t_c \times t_h \times t_w}$ via deep convolution layers; after that, $m_v$ is sent into a fully-connected layer with SoftMax to determine whether the input video is real or fake from a global perspective. We modify the DCNN in the following ways: 1) One 3D convolution layer replaces the last two linear layers, 2) We reduce the layer's size accordingly (for 64 $\times$ 64 videos) for better results and time complexity, 3) Following each 3D convolution, batch normalization, leakyReLU activation function, and max-pooling (down-sampling) layer is used, 4) Embedded text ($\varphi_t$) along with the extra CA "`c" variable are combined in the 2nd last layer. 5) The Sigmoid activation function is used at last. In more detail, convolution, batch-normalization, and LeakyReLU activation layers are employed first, followed by another convolution and down-pooling (Max-pooling) layer to downscale the frame size by a factor of two across consecutive levels of feature maps. This process is repeated several times $n = 5$ to down-sample the video, which contains 32 frames. Every step is a memory unit, so the state of these memory units is communicated between frames, allowing each frame to be built based on historical data and contributing to temporal coherence. Finally, we down-sample the video to a single vector to classify whether the video is real or generated.

The following objects can be used to jointly train the weights of the Generator (G) and Discrimination (D).

$$\overset{min}{\varnothing_g}\ \overset{max}{\varnothing_d}\ \mathbb{E}_{V\sim p_{data}(v)}\left[log\ D\left(V|\varphi_t + \text{`c}\ ;\ \varnothing_d\right)\right] + \mathbb{E}_{z\sim p(z)}\left[\log\left(1 - D\left(G\left(z + \text{`c}_t\ ;\ \varnothing_g\right)|\varphi_t + \text{`c}_t\ ;\ \varnothing_d\right)\right)\right] \tag{12}$$

where G and D are the generator and discriminator parameters, respectively, the distribution of actual videos $v$ is represented by $p_{data}(v)$. The $\varnothing_d$ and $\varnothing_g$ used for maintaining the maximum and minimum Lipschitz constants of the function; the noise taken from a Gaussian distribution is denoted by $z$, while the text condition is denoted by $\varphi_t$. When $pG(z) = p_{data}$ the object will reach the global optimum.

## 4 Experiments

This section presents the experimental details and results of our DD-GAN. To provide a comprehensive evaluation, we compared our proposed model with several state-of-the-art methods on three datasets: Mnist-4 single and two-digit moving/bouncing datasets (SBMG and TBMG) and a custom-generated dataset. Our evaluation included both qualitative and quantitative metrics. By doing so, we aimed to better understand and demonstrate the performance of our DD-GAN in comparison to existing methods.

### *4.1 Dataset Details and Creation*

First, we trained our proposed model on single-digit bouncing Mnist-4 and two-digit bouncing Mnist-4 datasets (SBMG and TBMG). Both datasets were publicly available and created according to Mittal et al. [48]. In these datasets, every video contains 16 frames, where every frame has exactly 64 $\times$ 64 pixels in size. Furthermore, there is no publically available dataset for preschool mathematics videos with associated text, so we used the same trick as Li et al. [21] and downloaded a huge number of videos from YouTube and Google with related text and tags, as well as some videos made by ourselves, and wrote the associated text for them. We used the similar concept of Kim et al. [45], who created a dataset for large-scale video classification. All these three datasets' details are given below.

#### *4.1.1 Single-Digit Bouncing Mnist-4 Dataset (SBMG)*

This is the publically available synthetic dataset. In this dataset, only one digit from 0–9 is moving/bouncing in a specific direction. This dataset contains 1200 GIFs; each GIF is a combination

of 16 frames, where every frame size is exactly 64 × 64 pixels. This dataset mainly focused on two motions, which are up-down and right-left. Every GIF is associated with a single text sentence describing the digits and their direction. We know that two motions are slightly simple, so we improved the dataset with more motion directions like; down-to-up, up-to-down, right-to-left, and left-to-right.

#### *4.1.2 Two-Digits Bouncing Mnist-4 Dataset (TBMG)*

This dataset is also a publicly available synthetic dataset. In this dataset, two digits from 0–9 are moving/bouncing together in similar or different directions. Two-digit bouncing Mnist-4 is a new and complicated version of the single-digit bouncing Mnist dataset. Similarly, this dataset also has 16 frames of GIFs with a similar exact 64 × 64-pixel size for a single frame. We also improved this dataset with similar motion directions: down-up, left-right, right-to-left, and left-to-right.

#### *4.1.3 Custom Generated Dataset (Math Dataset)*

For this research, we created a new preschool mathematics video dataset. We used the same trick as Li et al. [21] and downloaded videos from YouTube and Google with associated text and tags; we also made some videos by ourselves with related text descriptions. However, downloaded videos are too long, so we divided them into small parts of 3–10 s and later converted them into GIFs. Moreover, establish it at 64 × 64 pixels in 32 frames of each GIF. The processes we have followed up on for the data collection are as follows; we selected four keywords; "Addition," "Subtraction," "Multiplication," and "Division." After that, we downloaded videos for each keyword with their title, tags, duration, and description from YouTube and Google and made some by ourselves. Onward, outlier-removal techniques are used to clean the dataset and follow the Breg et al. [51] techniques to achieve the ten most frequent tags for the set of videos. The quality of the chosen tags is further ensured by correlating them to words in the ImageNet [52] and ActionBank [53] categories. These two datasets verify that the chosen tags contain visually natural objects and actions. Only videos that have at least three of the tags chosen were considered. Some other requirements for our dataset are: 1) Every video or GIF contains 3–10 s of duration. 2) The title and description are purely written in English. 3) The title has at least 4–5 meaningful words out of digits and stops words. 4) Our dataset contains 1040 videos or GIFs collected from different sources; every category has 200+ GIFs.

### *4.2 Implementation*

For robust comparison, we followed Pan et al.'s [22] methodology. As we discussed, we are only focused on 64 × 64 pixels of 32 frames of video generation from the text. So, $f_l = 32$ and $f_h = f_w = 64$. In the whole experiment, we used $n = 5$ steps, which means that we generated 32 frames of video. We used a bi-LSTM model for sentence encoding, where input, output, and hidden layers are all set to 256, i.e., $F = f_p = 256$. The random noise variable $z$ is set to 100 dimensions. The weights are all calculated using an independent Gaussian distribution $N\left(\mu\left(\varphi t\right), \sum\left(\varphi t\right)\right)$ and a normal distribution with a mean $\mu$ or average of 0 and a standard deviation $\sigma$ of 0.02. In leakyReLU, the slope of the leak is set at 0.2. Moreover, we set the mini-batch to 64 and the learning rate to 0.0002, as well as the momentum to 0.5. Onward, Generator (G) used up-pooling (up-sample) layers and the Tanh activation function. In contrast, Discriminator (D) used down-pooling (down-sample/max-pooling) layers and the sigmoid activation function to distinguish the real and fake frames. When training the DD-GAN model, we used the HP Pavilion series with 256 GB of Solid State Drive (SSD), a 1 TB hard drive, and an 8 GB NVIDIA GeForce GPU. The training time was several days while using this system.

The choice of VGG-16 is for DD-GAN because it has been widely used in image classification and recognition tasks and has achieved state-of-the-art performance on several benchmark datasets, especially the datasets used in this study. While ResNet-50 and U-Net are advanced models that have shown promising results in computer vision tasks, they were originally designed for different purposes. ResNet-50 for vanishing gradients and U-Net for image segmentation tasks. Therefore, their architectures may not be optimal for text-to-video generation tasks. On the other side, VGG-16 has been shown to extract high-level features from images, making it ideal for generating high-quality videos from text. Our experiments have proven that VGG-16 performs exceptionally well in this task, producing results on par with other state-of-the-art models.

### *4.3 Quantitative Evaluation*

This research must consider both the visual quality and the semantic match when generating a video from the text. For qualitative evaluation, we used four quantifiable evaluation metrics, which are: 1) Inception Score (IS), a mathematical measurement method proposed by Saliman et al. [54]. IS is used to observe the diversity and classification quality of the generated frames. 2) Fréchet Inception Distance (FID), this evaluation metric is used to evaluate the visual quality of each frame. 3) FID2vid, this metric is used for the temporal consistency and visual quality measurement of the whole video. Furthermore, we also used 4) Generative Adversarial Metric (GAM) further evaluates the generated videos. More details of these matrices are given below.

#### *4.3.1 Inception Score (IS)*

Judging the performance of a generative model is not easy. But, if we want to do that, some numerical approaches are used. One is the Inception Score (IS), a standard quantitative evaluation metric in generative models.

$$\mathrm{I} = \exp\left(E_s \mathrm{D}K\mathcal{L}\left(p\left(y|s\right) \,||\, p\left(y\right)\right)\right) \tag{13}$$

Here, $s$ represents the generated frames or samples, where $y$ is labeled as a predicated parameter of the inception model. The primary motivation for introducing this metric was to generate or synthesize meaningful frames that show the diversity and classification quality of the model. As a result, the $K\mathcal{L}$ divergence between the conditional probability distance $p\left(y|s\right)$ and the marginal probability distribution $p\left(y\right)$ must be significant. Fine-grained Mnist-4 single-digit bouncing, two-digits bouncing (SBMG and TBMS), and custom datasets are precisely adapted to the "sp" to get the best possible performance in the inception model. We tested the measure on many samples based on [54], with each model randomly picking over 1000 samples. The Inception Score (IS) results of our proposed method and other state-of-the-art methods can be seen in Table 1; a higher score means better performance. More evaluations based on Generative Adversarial Metric (GAM), which was proposed by Im et al. [55], are discussed later in this paper.

#### *4.3.2 Fréchet Inception Distance (FID)*

FID is the most common quantitative evaluation metric used in generative models, especially in GAN. This metric is used to evaluate the visual quality of the generated frames, in contrast to the previous Inception Score (IS), which considers the distribution of the generated frames. In contrast, the FID compares the distribution of the fake frames (generated by GAN) with the distribution of

real frames (real dataset frames) that were used to train the Generator (G). We can calculate the FID score by using the following education:

$$F^2 = ||f_{u\_1} - f_{u\_2}||^2 + T_r\left(c_{\_1} + c_{\_2} - 2 \times \sqrt{c}_{\_1} + c_{\_2}\right) \tag{14}$$

**Table 1:** Performance comparison of our DD-GAN with other state-of-the-art methods by FID, FID2vid, and Inception Score (IS). Smaller is better in FID and FID2vid, while higher is better in IS

| Method | SBMG | | | TBMG | | |
|---|---|---|---|---|---|---|
| | FID | FID2vid | IS | FID | FID2vid | IS |
| T2V [21] | 130.24 | 4.81 | 3.91 | 153.61 | 6.91 | 3.65 |
| TFGAN [32] | 47.76 | 7.19 | 3.96 | 55.71 | 7.70 | 3.84 |
| Sync-DRAW [48] | 69.75 | 4.54 | — | 101.87 | 5.26 | — |
| VGAN [28] | 170.31 | 6.59 | — | 168.64 | 5.97 | — |
| TGAN-C [22] | 63.05 | 4.84 | 4.71 | 57.59 | 5.36 | 4.83 |
| GAN-CLS [16] | 252.74 | 4.59 | — | 247.28 | 6.19 | — |
| BoGAN [20] | 47.57 | 3.12 | 5.05 | 48.31 | 4.22 | 5.01 |
| MoCoGAN [30] | 89.14 | 4.66 | 4.98 | 95.01 | 5.54 | 4.90 |
| Cap2vid [49] | 40.38 | 3.31 | — | 53.06 | 5.22 | — |
| IRC-GAN [56] | 71.51 | 4.57 | 4.99 | 80.13 | 5.14 | 4.95 |
| DD-GAN **(Ours)** | 38.21 | 3.03 | 5.51 | 41.18 | 4.08 | 5.46 |

The score is denoted by $F^2$, indicating that it is a distance measurement using square units. The feature-wise mean of the real frames and generated frames are referred to as $f_{u\_1}$ and $f_{u\_2}$, respectively. Here $c_{\_1} + c_{\_2}$ the covariance matrices of the actual feature vector and the generated feature vector, commonly denoted by Sigma ($\Sigma$). The $| |f_{u\_1} - f_{u\_2}| |^2$ means the sum squared difference between the two mean vectors, where $T_r$ denotes the trace linear algebra operation. We can see the FID score in Table 1 and the FID feature distance in Fig. 5.

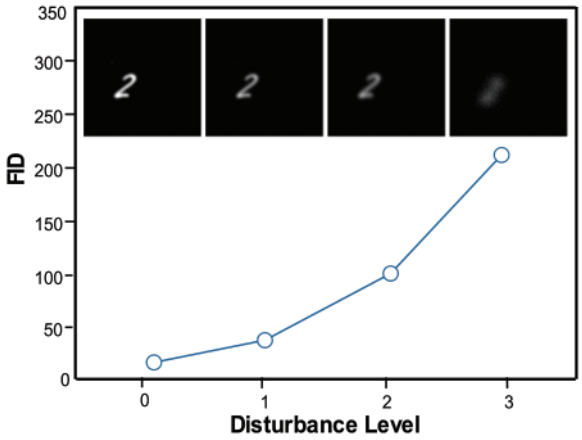


**Figure 5:** FID score calculating using the SBMG dataset

#### *4.3.3 Fréchet Inception Distance to Video (FID2vid)*

It is an updated form of FID; FID is a frame- or image-level comparison, while FID2vid is a video-level comparison. We extracted features of the 2nd to the last layer from the Generator (G), where G is trained on SBMG and TBMG datasets. Between the real videos and the generated videos, an FID score is calculated. This metric is used for the visual quality and temporal consistency measurement of the whole video.

#### *4.3.4 Generative Adversarial Matric (GAM)*

This metric was proposed by Im et al. [55]. GAM allows you to compare two generative adversarial models by competitively pitting one against the other. Given the two generative models:

$$Model_1 = (G_1,\ D_1)\ and\ Model_2 = (G_2,\ \ D_2) \tag{15}$$

The ratio of the two types between the discriminators of the two models is calculated.

$$y_{test} = \frac{\in (D_1 (x_{test}))}{\in (D_2 (x_{test}))},\ y_{sample} = \frac{\in (D_1 (G_2 (z)))}{\in (D_2 (G_1 (z)))} \tag{16}$$

The average classification error rate is denoted by $\in$ (), while the testing set is represented by $x_{test}$. If the $y_{test}$ is close to 1, meaning the two models have almost equal capacity to identify the real videos. The connection between $y_{sample}$ and 1 can tell which model is more likely to deceive the other model. For more details, see [55]. We change the mutual information term's weights to ensure that the $y_{test}$ is near to 1, which allows us to compare the two models using the $y_{sample}$. The results of GAM can be seen in Table 2.

**Table 2:** Generative Adversarial Metric (GAM): The $y_{sample}$ score with $y_{test}$ balanced to one

| Battle | SBMG | TBMG |
|---|---|---|
| DD-GAN *vs.* TGAN-C [22] | 0.873 | 0.763 |
| DD-GAN *vs.* IRC-GAN [56] | 0.632 | 0.581 |
| DD-GAN *vs.* VGAN [28] | 0.790 | 0.602 |
| DD-GAN *vs.* MoCoGAN [30] | 0.512 | 0.498 |
| DD-GAN *vs.* BoGAN [20] | 0.573 | 0.510 |
| DD-GAN *vs.* GAN-CLS [16] | 0.863 | 0.629 |

### *4.4 Human Rank (HR) or Human Evaluation*

To better evaluate our proposed DD-GAN model, we also conducted human studies to assess the generated videos' visual quality and semantic consistency. That is because IS, FID, FID2vid, and GAM focus only on measuring the realism of the generated videos. Still, on the other side, they all ignore the semantic match between the generated video and the text description. We followed Chen et al.'s [20] methodology. Forty generated videos with associated text descriptions from the Mnist-4 (SBMG) and 40 from the Mnist-4 (TBMG) were randomly selected and evaluated by 80 human subjects, which are university and college students from Abdul Wali Khan University Mardan (Timergara Campus) and Government Post Graduate College Timergara Dir Lower, KPK, Pakistan. Students rate the generated videos according to three criteria, which are: 1) Realism: This means how much realism is found in the generated videos according to the given text. 2) Relevance: This

means matching between the generated videos and their text description. 3) Coherence: This means the integrity of consecutive frames or the consistency of time across multiple frames.

Each criterion has ten rankings, ranging from 1 to 10, with 1 for bad and 10 for good. After collecting all the data, our proposed DD-GAN model achieved better performance as compared to other state-of-the-art models. The result of the human study can be seen in Table 3.

**Table 3:** Our proposed DD-GAN method averages ratings on each criterion of all created videos by each approach on the SBMG and TBMG datasets (realism, relevance, and coherence). Higher (better)

| Method | Realism | Relevance | Coherence |
|---|---|---|---|
| T2V [21] | 4.21 | 4.33 | 4.78 |
| Sync-DRAW [48] | 3.47 | 3.48 | 4.17 |
| GAN-CLS [16] | 5.69 | 5.80 | 6.06 |
| VGAN [28] | 4.53 | 4.54 | 4.98 |
| BoGAN [20] | 7.52 | 7.76 | 8.52 |
| Cap2vid [49] | 4.40 | 4.59 | 4.42 |
| TGAN-C [22] | 4.87 | 4.97 | 5.35 |
| DD-GAN **(Ours)** | 7.80 | 7.91 | 8.42 |

### *4.5 Compared Methods*

Several state-of-the-art methods are used to compare the performance of our DD-GAN method. Those comparison methods are: GAN Conditional Latent Space (GAN-CLS) [16], BoGAN [20], T2V [21], TGAN-C [22], VGAN [28], MocoGAN [30], TFGAN [32], Sync-DRAW [48], Cap2vid [49], and IRC-GAN [56]. The comparison results are shown in Figs. 6 and 7, as well as Tables 1–3.


**Figure 6:** Using the TBMG dataset, the experimental results of our DD-GAN and various other approaches for the caption "digit 7 is going right then left while digit 3 is going down than up"

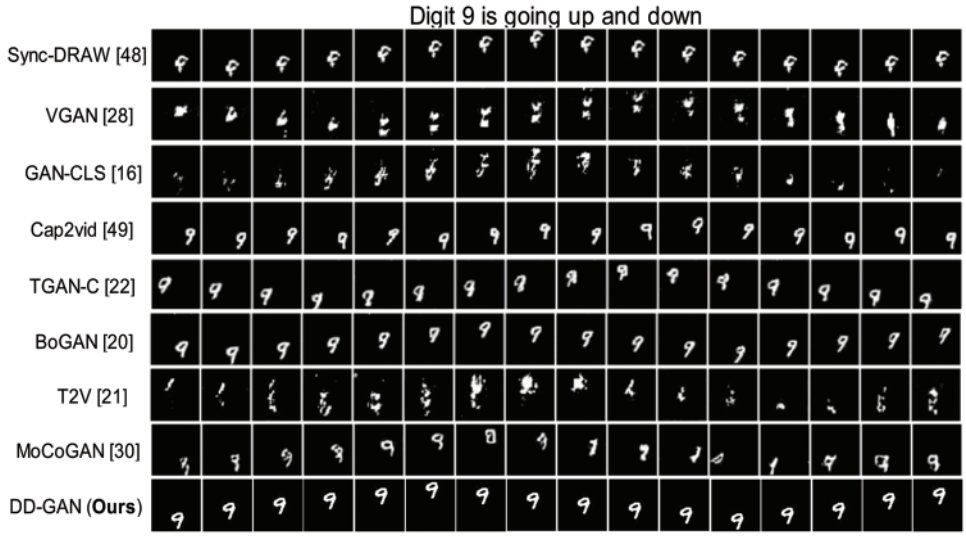


**Figure 7:** Using the SBMG dataset, the experimental results of our DD-GAN and various other approaches for the caption "digit 9 is going up and down"

### *4.6 Qualitative Analysis and Results*

Figs. 6 and 7 show examples of results generated by several models, including our DD-GAN using Mnist-4 single-digit bouncing (SBMG) and two-digit bouncing (TBMG) datasets. VGAN-c doesn't converge and seems to perform poorly. Although Cap2vid achieves the best FID score on the Mnist-4 single-digit moving (SBMS) dataset for synthetic data, it cannot capture the coherence between individual frames, causing the visual output of this method to appear disordered over the sequence of the produced videos. Regarding temporal organization, the output of TGAN-C, IRC-GAN, and Sync-DRAW appears well-organized, but every frame's output digit is deformed. Although the results generated using GAN-CLS appear realistic, they have several noises in each frame and lack movement between frames. In a photo-realistic example, IRC-GAN, MoCoGAN, VGAN-C, and TGAN-C get some convincing but blurred outputs related to the input description. BoGAN achieved some excellent results using Mnist-4 (SMBG and TBMG) but faces some issues in the temporal coherence between frames. Ultimately, our DD-GAN generated competitive results using the Mnist-4 (SBMG and TBMG) datasets. Our proposed method results show that our model can generate videos of clear, fine quality, and coherence. Using a custom-generated dataset, our model can also generate preschool mathematics videos from the associated text, and the results are exciting and impressive.

For more qualitative evaluation, we presented a lot of additional generated intact samples, which can be seen in Figs. 8–10. The results of our proposed DD-GAN method using the Mnist-4 single-digit bouncing (SBMG) dataset are shown in Fig. 10, and Mnist-4 two-digit bouncing (TBMG) results can be seen in Fig. 8 while using a custom generated dataset, the results are shown in Fig. 9. When compared to videos generated using other approaches, the videos generated by our DD-GAN look more realistic and more similar to the real videos.

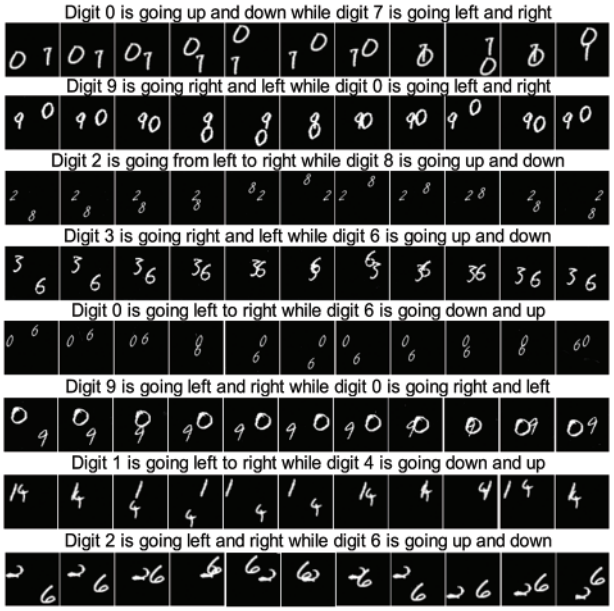


**Figure 8:** Generated videos from text descriptions by our DD-GAN using the TBMG Dataset


**Figure 9:** Generated videos from text descriptions by our DD-GAN using our custom Math dataset

### *4.7 More Discussion and Limitations*

#### *4.7.1 More Results on High-Resolution Video*

The proposed research is not limited to 64 × 64 resolutions. We increased the $n = 6$ to generate 128 × 128 pixels of resolution videos, and the results were quite comparative.

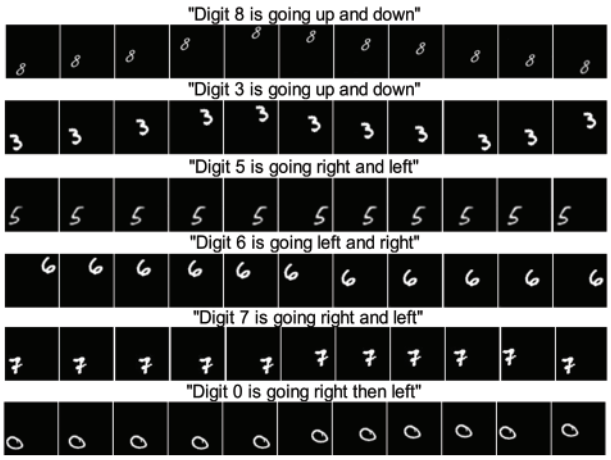


**Figure 10:** Generated videos from text descriptions by our DD-GAN using the SBMG dataset

### *4.7.2 Generating More Video Frames*

The proposed research generated 16 and 32 frames of video. Onward, we also tried to generate more frames for judging the evaluation of our model, and we did it successfully, but the coherence between frames needed to be greater.

### *4.7.3 No Motion Case Videos*

First, we tried to insert a sentence without including the movement and directions of the digits, for example, "Digit 2 + digit 1 = digit 3". We found that our model generated a video without any movement of the digits. After that, we inserted a sentence including digit motion and direction, for example, "Digit 2 + digit 2 = digit 4 moving from up to down", and then, as a result, generated a video with digit 4 moving from up to down. Fig. 11 shows the results of the custom dataset without moving any digits.

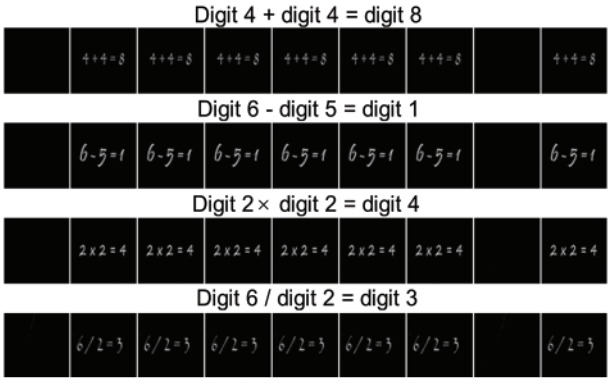


**Figure 11:** Generated without motion videos from text descriptions by our DD-GAN using the custom-generated dataset (no motion case)

#### *4.7.4 Failure Case Videos*

In some cases, we faced some failures; when we inserted complex, multiple entities, and long text description sentences in that case, we found that our proposed model (DD-GAN) did not work fine because the semantic information was difficult to extract. Moreover, our custom-generated dataset is also complicated, so sometimes the generated videos contain disconnectivity between frames.

#### *4.7.5 Complex Structure Limitations*

Some more limitations can be found in our proposed DD-GAN, such as instability when learning over too many frames, the inability to generate well-defined videos of larger size as well as of longer length, and the inability to insert complex and long description sentences, and the results can be affected. Moreover, while generating essential mathematics videos from text, some limitations need to be addressed. These include the fact that the movement of digits is not permanently fixed. It is just moving in the same direction again and again. Additionally, giving motion to all digits can result in confusing and unintelligible video content. The direction of digit movements can also be problematic, among other issues. While these limitations present significant challenges, they will be considered in future work to improve the quality and clarity of generated mathematics videos. In our future work, we are considering training DD-GAN better to generate longer, more extended in size, and more realistic frame videos from a text description, including but not limited to mathematics videos.

#### *4.7.6 DD-GAN Step-by-Step Improvement*

The proposed DD-GAN has undergone several improvements to enhance its efficiency and accuracy. Initially, we utilized a traditional deep convolutional GAN, but we gradually enhanced the performance by incorporating advanced techniques. We first integrated an LSTM text encoder into our model, which resulted in a slight improvement in accuracy. Next, we implemented Conditioning Augmentation (CA) techniques, changing layers, reducing the number of layers, and experimenting with different activation functions. These incremental changes allowed us to improve efficiency and accuracy significantly. Please refer to Table 4, Figs. 12 and 13 to see the step-by-step process and the corresponding performance gains. In the table, we used keywords with different meanings, such as NM: No Modification, LSTM: Long Short Term Memory, CA: Conditioning Augmentation, CL: Changing Layers, RLS: Reducing Layer Size, AF: Activation functions, and MM: More Modification.

**Table 4:** The effect of every modification of the DD-GAN is evaluated on the SBMG and TBMG datasets. A high inception score is better, while low FID and FID2vid mean better results

| Method | SBMG dataset | | | TBMG dataset | | |
|---|---|---|---|---|---|---|
| | IS | FID | FID2vid | IS | FID | FID2vid |
| DCGAN NM | 3.71 | 130.23 | 4.91 | 3.57 | 135.84 | 5.36 |
| DCGAN NM + LSTM | 3.98 | 105.74 | 4.26 | 3.87 | 111.59 | 5.03 |
| DCGAN NM + LSTM + CA | 4.23 | 83.19 | 3.91 | 4.16 | 87.16 | 4.81 |
| LSTM + CA + CL | 4.51 | 68.89 | 3.69 | 4.43 | 71.24 | 4.60 |
| LSTM + CA + RLS | 4.87 | 52.47 | 3.46 | 4.78 | 55.81 | 4.39 |
| LSTM + CA + AF | 5.31 | 41.53 | 3.19 | 5.22 | 44.62 | 4.21 |
| LSTM + CA + MM (final) | 5.51 | 38.21 | 3.03 | 5.46 | 41.18 | 4.08 |

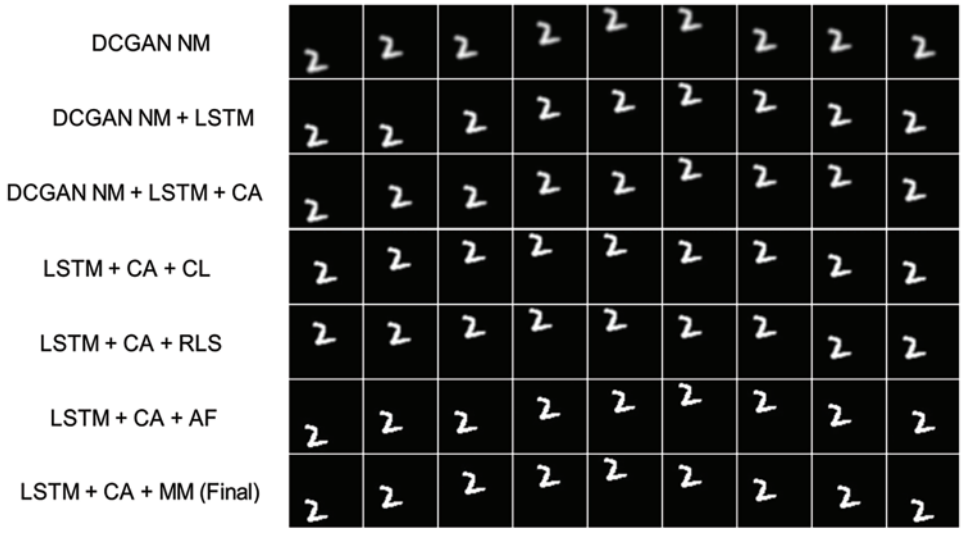


**Figure 12:** The effect of every modification of the DD-GAN evaluated on the TBMG dataset


**Figure 13:** The effect of every modification of the DD-GAN evaluated on the SBMG dataset

## 5 Conclusion

Generating realistic and synthetic videos from a text description is a complex task, requiring sophisticated techniques. This research proposes a Deep Deconvolutional Generative Adversarial Network (DD-GAN) framework to address this challenge. The DD-GAN framework consists of a Deep Deconvolutional Neural Network (DDNN) as a Generator (G), which generates moving digit videos from the text. The generated videos cannot be distinguished from real ones by a Deep Convolutional Neural Network (DCNN) used as a Discriminator (D). The DD-GAN model can generate moving digits and preschool math videos while maintaining temporal coherence between adjacent frames and ensuring that the resulting videos are well-matched with the given text. To train the DD-GAN model, we used two publicly available synthetic Mnist datasets (SBMG and TBMG), a custom dataset of mathematics videos that we collected from publicly accessible online

sources, and some videos that were custom generated with matching text-video pairs. The proposed model performed well based on evaluation metrics, including Inception Score (IS), FID, FID2vid, and Generative Adversarial Metric (GAM), compared to existing state-of-the-art GAN methods. Similarly, our proposed DD-GAN also performs well regarding realism, relevance, and coherence based on human studies. Moreover, we discussed future directions and limitations that can be addressed soon.

**Acknowledgement:** The authors are thankful to the National Engineering Research Center for E-Learning at Central China Normal University (Wuhan) and Wollongong Joint Institute Central China Normal University (Wuhan) for every bit of support. We would like to extend our acknowledgement to those who are involved in this work, directly or indirectly.

**Funding Statement:** This work is partially supported by the General Program of the National Natural Science Foundation of China (Grant No. 61977029).

**Author Contributions:** A. Ullah: Conceptualization, methodology, software, review, editing, writing original draft and funding acquisition. X. Yu: Conceptualization, methodology, reviewing and funding acquisition. M. Numan: Validation, data collection and review.

**Availability of Data and Materials:** Data will be made available on request.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao *et al.,* "A survey on deep learning: Algorithms, techniques, and applications," *ACM Computer Survey*, vol. 51, no. 5, pp. 1–36, 2018.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley *et al.,* "Generative adversarial nets," in *Proc. of Advances in Neural Information Processing Systems 27 (NeurIPS)*, Montreal, Quebec, Canada, pp. 2672–2680, 2014.

[3] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng *et al.,* "Generative adversarial networks: Introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[4] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[5] A. Radford, L. Metz and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *4th Int. Conf. on Learning Representations (ICLR)*, San Juan, Puerto Rico, 2016.

[6] L. J. Ratliff, S. A. Burden and S. S. Sastry, "Characterization and computation of local Nash equilibria in continuous games," in *51st Annual Allerton Conf. on Communication, Control, and Computing (Allerton)*, Monticello, IL, USA, pp. 917–924, 2013.

[7] A. K. Cherian and E. Poovammal, "A novel alphasrgan for underwater image super resolution," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 1537–1552, 2021.

[8] K. Fu, J. Peng, H. Zhang, X. Wang and F. Jiang, "Image super-resolution based on generative adversarial networks: A brief review," *Computers, Materials & Continua*, vol. 64, no. 3, pp. 1977–1997, 2020.

[9] D. K. Park, S. Yoo, H. Bahng, J. Choo and N. Park, "Megan: Mixture of experts of generative adversarial networks for multimodal image generation," in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, pp. 878–884, 2018.

[10] C. Zhang and Y. Peng, "Visual data synthesis via gan for zero-shot video classification," in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, pp. 1128–1134, 2018.

[11] T. Zhang, Z. Zhang, W. Jia, X. He and J. Yang, "Generating cartoon images from face photos with cycle-consistent adversarial networks," *Computers, Materials & Continua*, vol. 69, no. 2, pp. 2733–2747, 2021.
[12] A. A. Mahmoud, H. A. Sayed and S. S. Mohamed, "Variant Wasserstein generative adversarial network applied on low dose CT image denoising," *Computers, Materials & Continua*, vol. 75, no. 2, pp. 4535–4552, 2023.
[13] X. Mao and Q. Li, "Unpaired multi-domain image generation via regularized conditional GANs," in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, pp. 2553–2559, 2018.
[14] G. Y. Hao, H. X. Yu and W. S. Zheng, "MIXGAN: Learning concepts from different domains for mixture generation," in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, pp. 2212–2219, 2018.
[15] Y. Tian, X. Peng, L. Zhao, S. Zhang and D. N. Metaxas, "CR-GAN: Learning complete representations for multi-view generation," in *Proc. of 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Stockholm, Sweden, pp. 942–948, 2018.
[16] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele *et al.,* "Generative adversarial text to image synthesis," in *Proc. of 33rd Int. Conf. on Machine Learning*, New York, USA, vol. 48, pp. 1060–1069, 2016.
[17] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan *et al.,* "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 1316–1324, 2018.
[18] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang *et al.,* "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 5908–5916, 2017.
[19] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang *et al.,* "Stackgan++: Realistic image synthesis with stacked generative adversarial networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1947–1962, 2018.
[20] Q. Chen, Q. Wu, J. Chen, Q. Wu, A. V. D. Hengel *et al.,* "Scripted video generation with a bottom-up generative adversarial network," *IEEE Transactions on Image Processing*, vol. 29, pp. 7454–7467, 2020.
[21] Y. Li, M. Min, D. Shen, D. Carlson and L. Carin, "Video generation from text," in *Proc. of the AAAI Conf. on Artificial Intelligence*, New Orleans, Louisiana, USA, vol. 32, no. 1, 2018.
[22] Y. Pan, Z. Qiu, T. Yao, H. Li and T. Mei, "To create what you tell: Generating videos from captions," in *Proc. of the 25th ACM Int. Conf. on Multimedia*, Mountain View, CA, USA, pp. 1789–1798, 2017.
[23] D. Kim, D. Joo and J. Kim, "TiVGAN: Text to image to video generation with step-by-step evolutionary generator," *IEEE Access*, vol. 8, pp. 153113–153122, 2020.
[24] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. of Advances in Neural Information Processing Systems 30 (NIPS)*, California, USA, pp. 6626–6637, 2017.
[25] T. C. Wang, M. Y. Liu, J. Y. Zhu, G. Liu, A. Tao *et al.,* "Video-to-video synthesis," in *Proc. of Advances in Neural Information Processing Systems 31 (NeurIPS 2018)*, Montreal, Canada, pp. 1–14, 2018.
[26] D. J. Im, R. Memisevic, C. D. Kim and H. Jiang, "Generative adversarial metric," in *Int. Conf. on Learning Representation (ICLR)*, Caribe Hilton, San Juan, Puerto Rico, 2016.
[27] K. Lalit and D. K. Singh, "A comprehensive survey on generative adversarial networks for synthesizing multimedia content," *Multimedia Tools and Applications*, vol. 82, pp. 1–40, 2023.
[28] C. Vondrick, H. Pirsiavash and A. Torralba, "Generating videos with scene dynamics," in *30th Conf. on Neural Information Processing Systems (NIPS)*, Barcelona, Spain, pp. 613–621, 2016.
[29] N. Aldausari, A. Sowmya, N. Marcus and G. Mohammadi, "Video generative adversarial networks: A review," *ACM Computing Surveys (CSUR)*, vol. 55, no. 2, pp. 1–25, 2022.
[30] S. Tulyakov, M. Liu, X. Yang and J. Kautz, "MoCoGAN: Decomposing motion and content for video generation," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 1526–1535, 2018.

[31] Y. Li, Z. Gan, Y. Shen, J. Liu, Y. Chen *et al.,* "StoryGAN: A sequential conditional GAN for story visualization," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 6322–6331, 2019.
[32] 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 *Proc. of Twenty-Eighth Int. Joint Conf. on Artificial Intelligence (IJCAI)*, Macao, China, pp. 1995–2001, 2019.
[33] B. Zhang, M. He, J. Liao, P. V. Sander, L. Yuan *et al.,* "Deep exemplar-based video colorization," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 8044–8053, 2019.
[34] H. Dong, X. Liang, X. Shen, B. Wu, B. C. Chen *et al.,* "FW-GAN: Flow-navigated warping GAN for video virtual try-on," in *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Seoul, Korea (South), pp. 1161–1170, 2019.
[35] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn *et al.,* "Stochastic adversarial video prediction," in *Proc. of the Seventh Int. Conf. on Learning Representation (ICLR)*, New Orleans, LA, USA, 2019.
[36] T. Li, M. Slavcheva, M. Zollhoefer, S. Green, C. Lassner *et al.,* "Neural 3D video synthesis from multi-view video," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, pp. 5511–5521, 2022.
[37] T. C. Wang, M. Y. Liu, A. Tao, G. Liu, J. Kautz *et al.,* "Few-shot video-to-video synthesis," in *Proc. of 33rd Int. Conf. on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, pp. 5013–5024, 2019.
[38] Y. Zeng, J. Fu and H. Chao, "Learning joint spatial-temporal transformations for video inpainting," in *European Conf. on Computer Vision (ECCV)*, Springer, Cham, pp. 528–543, 2020. https://doi.org/10.1007/978-3-030-58517-4_31
[39] Y. Chang, Z. Liu, K. Lee and W. Hsu, "Free-form video inpainting with 3D gated convolution and temporal PatchGAN," in *IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Seoul, Korea (South), pp. 9065–9074, 2019.
[40] N. Saleem, J. Gao, M. Irfan, E. Verdu and J. P. Fuente, "E2e-v2SResNet: Deep residual convolutional neural networks for end-to-end video driven speech synthesis," *Image and Vision Computing*, vol. 119, pp. 104389, 2022.
[41] G. Mittal and B. Wang, "Animating face using disentangled audio representations," in *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, Snowmass Village, CO, USA, pp. 3279–3287, 2020.
[42] L. Chen, Z. Li, R. K. Maddox, Z. Duan and C. Xu, "Lip movements generation at a glance," in *European Conf. on Computer Vision (ECCV)*, Seoul, Korea, Springer, Cham, pp. 520–535, 2018.
[43] S. A. Jalalifar, H. Hasani and H. Aghajan, "Speech-driven facial reenactment using conditional generative adversarial networks," in *15th European Conf. on Computer Vision (ECCV), Lecture Notes in Computer Science*, Munich, Germany, vol. 11218, 2018.
[44] Y. Zhou, Z. Wang, C. Fang, T. Bui and T. Berg, "Dance dance generation: Motion transfer for internet videos," in *IEEE/CVF Int. Conf. on Computer Vision Workshop (ICCVW)*, Seoul, Korea (South), pp. 1208–1216, 2019.
[45] H. Kim, P. Garrido, A. Tewari, W. Xu, J. Thies *et al.,* "Deep video portraits," *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–14, 2018.
[46] L. Liu, W. Xu, M. Zollhofer, H. Kim and F. Bernard, "Neural rendering and reenactment of human actor videos," *ACM Transactions on Graphics*, vol. 38, no. 5, pp. 1–14, 2019.
[47] O. Gafni, L. Wolf and Y. Taigman, "Vid2Game: Controllable characters extracted from real-world videos," in *Proc. of the Eighth Int. Conf. on Learning Representation (ICLR-2019)*, Virtual Conference, 2019.
[48] G. Mittal, T. Marwah and V. Balasubramanian, "Sync-DRAW: Automatic video generation using deep recurrent attentive architectures," in *Proc. of 25th ACM Int. Conf. on Multimedia*, New York, USA, pp. 1096–1104, 2017.
[49] T. Marwah, G. Mittal and V. Balasubramanian, "Attentive semantic video generation using captions," in *IEEE Int. Conf. on Computer Vision (ICCV-2017)*, Venice, Italy, pp. 1435–1443, 2017.

[50] M. Arjovsky, S. Chintala and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. of 34th Int. Conf. on Machine Learning*, Sydney, Australia, vol. 70, pp. 214–223, 2017.
[51] T. L. Berg, A. C. Berg and J. Shih, "Automatic attribute discovery and characterization from noisy web data," in *Proc. of 11th European Conf. on Computer Vision*, Heraklion, Crete, Greece, Springer, pp. 663–676, 2010.
[52] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li *et al.,* "ImageNet: A large-scale hierarchical image database," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops (CVPR Workshops)*, Miami, FL, USA, pp. 248–255, 2009.
[53] J. Corso and S. Sadanand, "Action bank: A high-level representation of activity in video," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, pp. 1234–1241, 2012.
[54] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford *et al.,* "Improved techniques for training gans," in *Proc. of 30th Conf. on Neural Information Processing Systems (NIPS)*, Barcelona, Spain, pp. 2234–2242, 2016.
[55] D. J. Im, C. D. Kim, H. Jiang and R. Memisevic, "Generating images with recurrent adversarial networks," arXiv preprint arXiv:1602.05110, 2016.
[56] K. Deng, T. Fei, X. Huang and Y. Peng, "IRC-GAN: Introspective recurrent convolutional GAN for text-to-video generation," in *Proc. of Twenty-Eighth Int. Joint Conf. on Artificial Intelligence (IJCAI-19)*, Macao, China, pp. 2216–2222, 2019.