DCVGAN: DEPTH CONDITIONAL VIDEO GENERATION

Yuki Nakahira and Kazuhiko Kawamoto

Chiba University, Chiba, Japan

ABSTRACT

In the past few years, several generative adversarial networks (GANs) for video generation have been proposed although most of them only use color videos to train the generative model. However, to make the model understand scene dynamics more accurately, not only optical information but also three-dimensional geometrical information is important. In this paper, using depth video together with color video, we propose a GAN architecture for video generation. In the generator of our architecture, the depth video is generated in the first half and in the second half, the color video is generated by solving the domain translation from the depth to the color. By modeling the scene dynamics with a focus on the depth information, we were able to produce videos of higher quality than the conventional method. Furthermore, we show that our method produces better video samples than ones by conventional method in terms of both variety and quality when evaluating on facial expression and hand gesture datasets. The codes and generated sample videos are publicly available on Github[1].

Index Terms— video generation, generative adversarial nets, depth information, image translation

1. INTRODUCTION

Recently, automatic data generation using deep generative model has gained considerable attention owing to its useful applications e.g., content generation and data augmentation. In particular, the generative adversarial network known as GAN[1] demonstrates high accuracy on image generation tasks, such as high-resolution image generation, class conditional image generation and image domain translation. Therefore, the deep generative model has also been applied to video (which comprises a series of images) generation tasks, which are very important in the field of computer vision. However, since video generation require time series modeling, they are very challenging to handle.

In recent years GANs have been applied to video generation tasks as well. Vondrick et al. [2], who first applied GAN to video generation proposed video GAN (VGAN), and this method uses the knowledge that a video recorded by a fixed camera can be divided into two parts, namely, a moving foreground and a static background. In line to this, Saito et al. [3] pointed out that three-dimensional convolutional neural network (3DCNN) is inadequate for video generation tasks and instead proposed temporal GAN (TGAN), which uses a combination of one-dimensional CNN (1DCNN) and two-dimensional CNN (2DCNN). In this method, 1DCNN is used to convert a latent vector of a video into multiple latent vectors of images, and 2DCNN is used to generate images from latent codes. Ohnishi et al. [4] proposed optical flow and texture GAN (FTGAN) which generates optical flow from latent code, which is then used to generate motion plausible videos. Tulyakov et al. [5] suggested that a video can be decomposed into ”content” (elements that are consistently immutable) and ”motion” (elements that change over time), and proposed motion and content decomposed GAN (MoCoGAN) which generates a video from latent codes, each of which comprises ”content” and ”motion” parts.

However, the current state-of-art method, like the MoCoGAN, causes problems such as the extremely unnatural appearance of moving objects and assimilation of objects into the background. To generate more natural videos, it is necessary to make the model understand the region of the object clearly. Furthermore, the model should provide a natural motion to the said object, and simultaneously, consider the interaction between the object and other objects.

We believe that using only color videos is the underlying cause of aforementioned problems. Therefore, we argued that the geometrical information should be used for natural video generation in tandem with optical information of the scene. This idea is based on the following two points. (1) In the video, the motion is expressed by multiple images (2D planes) but the object essentially moves in three-dimensional space. Therefore, it can be said that it is difficult for the model to understand the scene dynamics accurately by using only two-dimensional optical information. (2) In the human brain, it is known that the depth of the scene is estimated using the two-dimensional optical information obtained from the retina. This suggests that the depth information plays a vital role in recognizing the real world correctly and is used to represent geometrical information of the scene.

Motivated by this idea, in this paper, we propose a GAN architecture that is capable of producing a higher quality color video using the depth information. The contributions of this

[1] https://github.com/raahii/dcvgan

978-1-5386-6249-6/19/$31.00 ©2019 IEEE

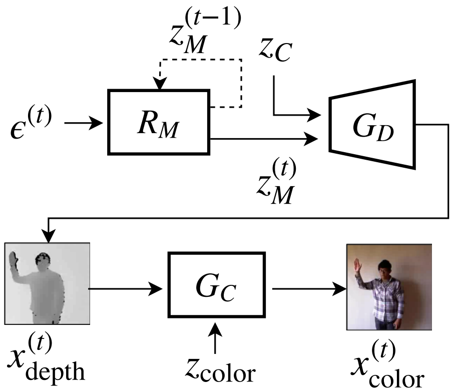


Fig. 1: Model architecture: The upper half is the depth video generation, and the bottom half is the domain translation from depth to color.

paper are as follows:

1. We propose a novel GAN architecture for video generation mapping noise vectors to color videos through depth videos.
2. To verify the effectiveness of the proposed method, we conduct experimental validation on two datasets with a subjective comparison to the state-of-the-art video generation method, MoCoGAN[5].

2. PROPOSED METHOD

The proposed method uses pairs of color video and depth video to train the generative model. Specifically, our architecture generates a video using the following two steps. (1) Generate the depth video to model the scene dynamics based on the geometrical information. (2) To add appropriate color to the geometrical information of the scene, the domain translation from depth to color is performed for each image. The architecture is shown in Figure 1.

2.1. Depth video generation

We designed a depth generation model in the first half of our generator based on MoCoGAN architecture, which decomposes the latent space into ”content” and ”motion”. This model is composed of two networks, frame seed generator (R_M) and depth image generator (G_D).

The frame seed generator is a recurrent neural network (RNN) models the dynamics of the video and generates motion latent vectors. Latent vectors corresponding to images of a video are represented as

$$\boldsymbol{z}^{(t)} = \begin{pmatrix} \boldsymbol{z}_C \\ \boldsymbol{z}_M^{(t)} \end{pmatrix},\ t = 1, 2, \ldots, T, \tag{1}$$

where T is the video length, $\boldsymbol{z_C}$ is a latent vector corresponding to the content of the video, which is sampled once and fixed in a video generation process to make content (shape and size) consistent within a video. We sample $\boldsymbol{z_C}$ from the Gaussian distribution with mean 0 and covariance matrix I_{d_C}, i.e.

$$\boldsymbol{z}_C \sim \mathcal{N}(0, \boldsymbol{I}_{d_C}) \tag{2}$$

where d_C is the dimension of latent vector $\boldsymbol{z_C}$, $\boldsymbol{I}_{d_C}$ is the $d_C \times d_C$ identity matrix. Conversely, $\boldsymbol{z}_M^{(t)}, t = 1, 2, ..., T$ are motion latent vectors corresponding to the motion of the video, they are generated by

$$\boldsymbol{z}_M^{(t)} = R_M(\boldsymbol{z}_M^{(t-1)}, \boldsymbol{\epsilon}^{(t)}),\ t = 1, 2, \ldots, T, \tag{3}$$

where $\boldsymbol{\epsilon}^{(t)} \sim \mathcal{N}(0, \boldsymbol{I}_{d_M})$ and d_M is the dimension of the latent vector z_M.

Depth image generator (G_D) is a CNN that generates a depth image from a latent vector using transposed convolution. Using the latent vectors in (1), our model generates a depth video by

$$\boldsymbol{x}_{\text{depth}}^{(t)} = G_D(\boldsymbol{z}^{(t)}), t = 1, 2, ..., T. \tag{4}$$

2.2. Color video translation

We designed a domain translation model from color to depth in the second half of the proposed generator based on pix2pix[6] architecture. The model is named color image generator (G_C), which is an encoder-decoder architecture that uses U-Net [7]. Domain translation is performed by each image, as

$$\boldsymbol{x}_{\text{color}}^{(t)} = G_C\left(\boldsymbol{x}_{\text{depth}}^{(t)}, \boldsymbol{z}_{\text{color}}\right), t = 1, 2, ..., T, \tag{5}$$

where $\boldsymbol{z}_{\text{color}} \sim \mathcal{N}(\boldsymbol{z}|0, \boldsymbol{I}_{\boldsymbol{d}_{\text{CR}}})$, $\boldsymbol{z}_{\text{color}}$ is a latent vector corresponding to output color scene. By using a common vector when converting images of a video, the latent vector keeps the consistency of the output video scene, and changes the output stochastically. We integrate $\boldsymbol{z}_{\text{color}}$ by concatenating with bottleneck feature maps outputted by the encoder.

2.3. Discriminators

Similar to MoCoGAN, our method has two discriminators: image discriminator and video discriminator. The image discriminator takes a pair of color image and depth image, which is randomly selected in a video and evaluates whether it is a sample came from the dataset or the generator. Conversely, the video discriminator takes a pair of color video and depth video as input, and evaluates it including its temporal features. Using two discriminators simultaneously to train the generator significantly improves the convergence of the adversarial training. [5]

The objective function of our model is defined by

$$\min_{R_M,G_D,G_C}\ \max_{D_I,D_V}\ V(R_M,G_D,G_C,D_I,D_V), \qquad (6)$$

$$\begin{aligned} & V(R_M,G_D,G_C,D_I,D_V) \\ & = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}}[\log D_I(\boldsymbol{x}^{(t)}_{\text{color}}, \boldsymbol{x}^{(t)}_{\text{depth}})] \\ & \quad + \mathbb{E}_{\tilde{\boldsymbol{x}}\sim p_g}[1-\log D_I(\tilde{\boldsymbol{x}}^{(t)}_{\text{color}}, \tilde{\boldsymbol{x}}^{(t)}_{\text{depth}})] \\ & \quad + \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}}[\log D_V(\boldsymbol{x}_{\text{color}}, \boldsymbol{x}_{\text{depth}})] \\ & \quad + \mathbb{E}_{\tilde{\boldsymbol{x}}\sim p_g}[1-\log D_V(\tilde{\boldsymbol{x}}_{\text{color}}, \tilde{\boldsymbol{x}}_{\text{depth}})], \end{aligned}$$

where $\boldsymbol{x}_{\text{color}}$, $\boldsymbol{x}_{\text{depth}}$ are dataset samples; $\tilde{\boldsymbol{x}}_{\text{color}}$, $\tilde{\boldsymbol{x}}_{\text{depth}}$ are generated samples; D_I is the image discriminator; D_V is the video discriminator; p_g is the distribution of the generator; and p_{data} is the distribution of the dataset.

3. EXPERIMENTS

We conducted experiments to evaluate our model. We compared our method with MoCoGAN [5], the state-of-the-art video generation method on human facial expression and hand gesture datasets. The details of the datasets are shown below. Furthermore, we processed both video datasets so that the image size is 64×64 and the video length is 16, and the pair of color video and depth video is used for the experiment.

- **Facial Expression**: We used MUG facial expression database [8] which records facial expressions of 86 people. Due to the fact that the dataset has only color videos, we created depth videos estimated by the three-dimensional face reconstruction method [9].
- **Hand Gesture**: We used ChaLearn LAP IsoGD dataset [10], which records various hand gestures of 21 people in both color and depth video format. Due to the diversity of the scene and hand movement, this is more complicated and challenging than facial expression.

The detailed design of each network is available on the Github page. Roughly, G_D consists of 5 transposed convolutional layers, and G_C consists of 7 convolutional layers and 7 transposed convolutional layers. D_I and D_V consists of five 2-dimensional or 3-dimensional convolutional layers, respectively. We used ADAM [11] for training, with a learning rate of 0.0002 and momentums of 0.5 and 0.999. To stabilize the training process, Gaussian noises with $\mu = 0, \sigma = 0.2$ are added to the input of each layer of the discriminator [12]. The dimension of the latent variables are $d_C = 40$, $d_M = 10$ and $d_{\text{CR}} = 10$.

Figure 2 shows the results of the generated samples. Using each generated sample, we compared our model and MoCoGAN by two quantitative evaluation metrics.

The first one is Fréchet inception distance (FID) [13], which calculates the distance between the set of generated samples and the set of dataset samples, and we observed that the smaller score, the better. The score is defined by

Table 1: Evaluation result on Fréchet inception distance [13].

	MoCoGAN	our model
Facial Expression	22.9 ± 0.289	$\mathbf{6.68 \pm 0.0699}$
Hand Gesture	87.2 ± 0.793	$\mathbf{25.3 \pm 0.364}$

$$\begin{aligned} \text{FID}(\mathbb{P}_r,\mathbb{P}_g) = & \ ||\boldsymbol{\mu_r} - \boldsymbol{\mu_g}||_2^2 + \\ & \text{Tr}(\boldsymbol{\Sigma_r} + \boldsymbol{\Sigma_g} - 2(\boldsymbol{\Sigma_r}\boldsymbol{\Sigma_g})^{\frac{1}{2}}), \end{aligned} \qquad (7)$$

where $\mathbb{P}_r$ is the distribution of the dataset; $\mathbb{P}_g$ is the distribution of the generator; $\boldsymbol{\mu_r}$ and $\boldsymbol{\Sigma_r}$ are mean vector and variance-covariance matrix of $\mathbb{P}_r$; $\boldsymbol{\mu_g}$ and $\boldsymbol{\Sigma_g}$ are mean vector and variance-covariance matrix of $\mathbb{P}_g$.

The second one is precision-recall distributions (PRD) [14], which was proposed to evaluate generated samples from two viewpoints, precision(quality) and recall(diversity). By drawing a PR curve based on the PRD results, we can determine the failure cases of the generated samples. The score is defined by

$$\text{PRD}(\mathbb{P}_r,\mathbb{P}_g) = \{(\alpha(\lambda),\beta(\lambda)) \mid \lambda \in \Lambda\} \qquad (8)$$

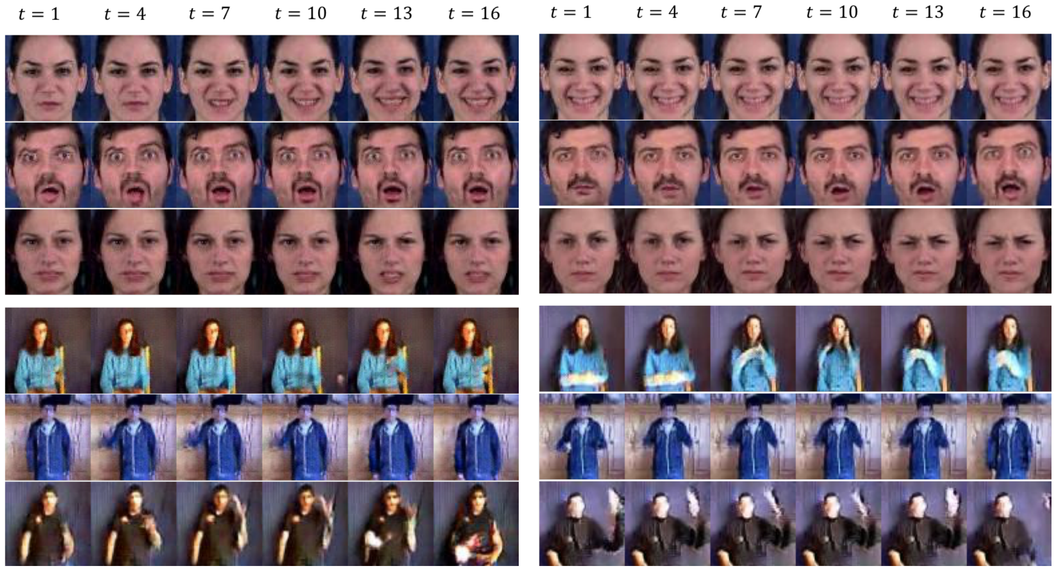
where

$$\begin{aligned} \Lambda &= \left\{\tan\left(\tfrac{i}{m+1}\tfrac{\pi}{2}\right) \mid i = 1,2,\ldots,m\right\}, \\ \alpha(\lambda) &= \sum_{\omega\in\Omega} \min\left(\lambda\mathbb{P}_r(\omega), \mathbb{P}_g(\omega)\right), \\ \beta(\lambda) &= \sum_{\omega\in\Omega} \min\left(\mathbb{P}_r(\omega), \mathbb{P}_g(\omega)/\lambda\right), \end{aligned}$$

$\mathbb{P}_r$ and $\mathbb{P}_g$ are probability distributions of the dataset and the generator defined on a finite state space Ω, $m \in \mathbb{R}$ is a given angular resolution which is set to 1001 in the experiment.

To calculate each metric, we used 3,000 samples for facial expression dataset and 10,000 samples for hand gesture dataset. It is known that using the convolutional feature of the intermediate layer obtained by the inception model as the representation of a video is better than using pixel space data to compute metric score[15]. Thus we calculated the convolutional feature for each generated sample before the evaluation using ResNet-101 which was previously trained on UCF-101[16] dataset.

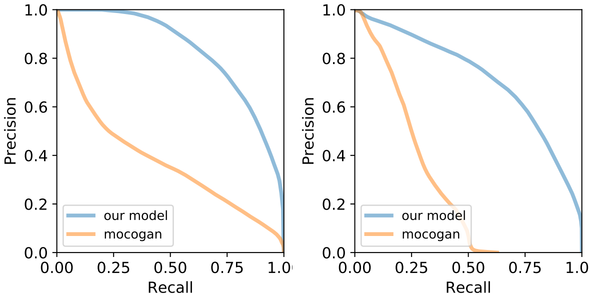
Table 1 shows the evaluation result on FID, and Figure 3 shows the evaluation result on PRD. From Table 1, it is shown that our model outperforms over MoCoGAN on both datasets because the scores of our model are smaller than ones of MoCoGAN. Moreover, from the PR curve in the Figure 3, the area under the curve of our model is clearly larger than that of MoCoGAN on both datasets. It indicates that our model outputted videos nicely balanced in quality and diversity.



(a) MoCoGAN

(b) our model

Fig. 2: Generated samples: the upper half is facial expression dataset, the lower half is hand gesture dataset.



(a) Facial Expression

(b) Hand Gesture

Fig. 3: Evaluation result on precision-recall distributions [14].

3.1. Discussion

Considering the two quantitative evaluation metrics, it is shown that the generated samples of our method produce better samples based on the viewpoints of quality and diversity than the MoCoGAN. Using the depth information, to model scene dynamics implies that the naturalness of motion and consistency of the scene are improved. In addition, the result of the facial expression dataset is also improved despite using the depth estimated by the three-dimensional face reconstruction method [9]. This result suggests that even predicted depth information is useful for the video generation task and it might be a solution to a situation needs high annotation cost.

Conversely, some improvements are found in the experimental results. In our architecture, color image generator used z_{color} to convert various color videos based on a depth video. However, when we convert the same depth video with a different z_{color}, the outputs become deterministic, which implies that they are simply ignored in the network. This problem is called one-to-one mapping problem and is caused by the training data that has only one-to-one pair samples. If the representation of depth and color can be clearly separated in the future, the diversity of generated samples will increase significantly and the effect of utilizing depth will be more enhanced.

4. CONCLUSION

In this paper, we proposed a method to generate videos by utilizing depth information, whereas most of the existing methods utilize only color information. Our model is inspired by the MoCoGAN but we newly design the network architecture in which the depth image generator and the domain translator are introduced. The experiments show that the proposed method outperforms MoCoGAN, which is the state-of-the-art model for video generation, and this result indicates that depth information is effective for generating high-quality videos.

5. ACKNOWLEDGEMENT

This work was supported by JSPS KAKENHI Grant Number JP16K00231 and JP19K12039.

6. REFERENCES

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Neural Information Processing Systems (NeurIPS)*, 2014, pp. 2672–2680.

[2] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba, "Generating videos with scene dynamics," in *Neural Information Processing Systems (NeurIPS)*, 2016, pp. 613–621.

[3] Masaki Saito, Eiichi Matsumoto, and Shunta Saito, "Temporal generative adversarial nets with singular value clipping," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2830–2839.

[4] Katsunori Ohnishi, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada, "Hierarchical video generation from orthogonal information: Optical flow and texture," in *AAAI Conference on Artificial Intelligence*, 2018.

[5] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz, "Mocogan: Decomposing motion and content for video generation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 1526–1535.

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros, "Image-to-image translation with conditional adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1125–1134.

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

[8] Niki Aifanti, Christos Papachristou, and Anastasios Delopoulos, "The mug facial expression database," in *Image Analysis for Multimedia Interactive Services (WIAMIS), 2010 11th International Workshop on*. IEEE, 2010, pp. 1–4.

[9] Aaron S Jackson, Adrian Bulat, Vasileios Argyriou, and Georgios Tzimiropoulos, "Large pose 3d face reconstruction from a single image via direct volumetric cnn regression," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1031–1039.

[10] Jun Wan, Yibing Zhao, Shuai Zhou, Isabelle Guyon, Sergio Escalera, and Stan Z Li, "Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2016, pp. 56–64.

[11] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[12] Martin Arjovsky and Léon Bottou, "Towards principled methods for training generative adversarial networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Neural Information Processing Systems (NeurIPS)*, 2017, pp. 6626–6637.

[14] Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly, "Assessing generative models via precision and recall," in *Neural Information Processing Systems (NeurIPS)*, 2018, pp. 5234–5243.

[15] Qiantong Xu, Gao Huang, Yang Yuan, Chuan Guo, Yu Sun, Felix Wu, and Kilian Q. Weinberger, "An empirical study on evaluation metrics of generative adversarial networks," *CoRR*, vol. abs/1806.07755, 2018.

[16] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *CRCV-TR*, 2012.