SSA-GAN: End-to-End Time-Lapse Video Generation with Spatial Self-Attention

Daichi Horita[1] and Keiji Yanai[1]

The University of Electro-Communications, Tokyo
{horita-d,yanai}@mm.cs.uec.ac.jp

Abstract. We usually predict how objects will move in the near future in our daily lives. However, how do we predict? In this paper, to address this problem, we propose a GAN-based network to predict the near future for fluid object domains such as cloud and beach scenes. Our model takes one frame and predict future frames. Inspired by the self-attention mechanism [25], we propose introducing the spatial self-attention mechanism into the model. The self-attention mechanism calculates the reaction at a certain position as a weighted sum of the features at all positions, which enables us to learn the model efficiently in one-stage learning. In the experiment, we show that our model is comparable compared with the state-of-the-art method which performs two-stage learning.

Keywords: Video Prediction · GAN · End-to-End training · Self-Attention

1 Introduction

Human beings predict how objects will move in the near future in daily life. However, how do we predict? Generally, we have gained a visual experience by looking at the relationships between objects interacting in various scenes of life. Based on the wealth of experience and knowledge gained through the experience, we predict the future in a few seconds later. In addition, we utilize the experience and knowledge when we encounter new scenes, and predict future movements. Therefore, predicting future movements plays an important role in intelligent systems and automatic systems as it is necessary to decide and plan how objects will move in the future. To achieve this, we need to train a model that understands how scenes change and how objects move.

Recently, in the field of computer vision, there are many studies using images and videos. In particular, solving the video problems is beneficial for various applications, such as video prediction [5,16], action recognition [12,27], action localization [7,22], video understanding [23,3], and video captioning [14,18]. In unsupervised learning of images, Generative Adversarial Nets (GANs) [1,6] show excellent results, and success in producing high resolution and quality images such as people's faces [13,20]. In contrast, in unsupervised learning of video, there are still difficult problems compared to the field of image generation. If

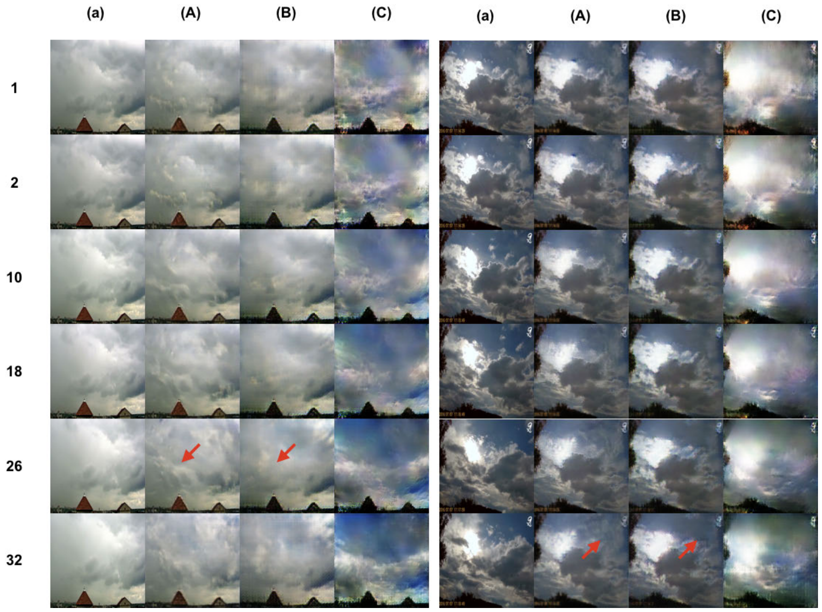

Fig. 1. Some example results of cloud motions generated by our proposed model trained with the cloud time-lapse dataset. From left to right: (a) the ground truth, (A) our model, (B) first stage of MD-GAN [29], and (C) second stage of MD-GAN, respectively. (B) and (C) are generated from official pre-trained models.

the model mistakenly learns the physical structure, the predicted motion may include movement which is physically impossible. In addition, the transition speed of scenes and movement speed of objects are also an important factor to make videos natural. Video prediction models have to predict appropriate speeds of objects and scenes as well as generate their appearances. Therefore, video prediction is a challenging problem and still needs much progress. In this paper, our goal is the future frame prediction of the video, which predicts natural movement according with human cognition.

Regarding video generation and prediction by neural networks, the studies started just four years ago. One of the representative works done in the early day is VGAN [26], which is a method that can generate moving images from large-scale unsupervised time-lapse video datasets. However, since VGAN generates the background and foreground of the image separately, the background is fixed, it has a problem that it can not generate the dynamic background. Therefore, it is necessary to learn a model that can generate not only the foreground but also the background at the same time. In addition, one of the biggest problems with video prediction is what we should focus on for training. The variation of motions in videos is very huge. It is unrealistic to generate many kinds of movements of any kind of object with the current techniques. Limiting video domains is a reasonable way to attack this problem. Even if one frame of consecutive moving images is given, it is difficult to set constraints to the model because there are multiple motion options for the future scene transition. As a result, frame prediction in unsupervised learning remains a very difficult problem. To achieve these issues, TemporalGAN [21] uses 2D convolution in the generator to handle temporal and spatial features separately. However, in contrast, we propose a method using 3D convolution to consider the problem of simultaneously dealing with temporal and spatial features.

Regarding learning methods, one-stage approaches are more efficient and elegant than two-stage methods because models are simple in general. However, in recent studies, two-stage training is prevailing. For example, MD-GAN [29] achieves excellent results with two-stages learning to predict future frames. In their method, they generate rough shapes and coarse dynamic movements in the first stage and add detailed appearances and motions to make the final video higher-quality in the second stage.

On the other hand, in this paper, we tackle a challenging problem with only one-stage learning. By introducing the spatial self-attention in the one-stage model in MD-GAN, we show that our model learns adequately in one-stage learning and predict more realistic future frames than MD-GAN, although the proposed model is simpler than existing two-stage methods.

Specifically, we propose the Spatial Self-Attention Generative Adversarial Network (SSA-GAN) model for future frame prediction. Our model consists of a generator and a discriminator. The generator has not a simple encoder-decoder architecture but the architecture like 3D U-Net [4] to avoid generating blurred images caused by losing content details. In addition, the generator has spacial self-attention layers based on [25] after each 3D convolutions and deconvolutions to preserve the spatial physical structure. Given a stationary input frame, the generator predicts future video frames which indicate how it will move in the future. In this way, our model keeps content details and predict as realistic dynamic scene transition as possible. We present a few example frames which are generated by our method and existing method. As shown in Fig. 1, the image frames generated by our model are realistic The red arrow indicates that our model can generate a more detailed image than the model before the introduction of this module.

Major contributions of this paper can be summarized as follows:

1. We propose the Spatial Self-Attention Generative Adversarial Networks(SSA-GAN) for video prediction.
2. We propose the spatial self-attention frameworks based on a self-attention mechanism [25], which enables our model to learn in one-stage while emphasizing spatial correlation between time series.
3. We introduce a model that sufficiently predicts future frames with an one-stage training and our model achieves comparable results with the state-of-the-art method.

2 Related Work

2.1 Generative Adversarial Networks

Generative adversarial networks (GANs) [1,6] have achieved impressive results in image generation [13,20] and image-to-image translation [11,34]. GANs consists of a generator and a discriminator. The discriminator learns to distinguish the produced fake samples from the real ones, while the generator learns to generate fake samples which are not distinguishable from the real ones. In this paper, we also leverage an adversarial loss to learn the mapping to generate future frames as realistic as possible.

2.2 Video generation

There are two main approaches to the field of video generation using GAN. One of them is to produce plausible videos by limiting video datasets to specific areas such as human faces and poses [2,30,32]. The other is a study to deal without such constraints [21,24,26]. MoCoGAN [24] generates videos efficiently by decomposing the latent space into content and motion subspaces. TemporalGAN [21] uses 2D convolution to generate video in the generator in order to handle temporal and spatial features separately. In this paper, our study is close to the latter because our model generates video frames with free movement without such constraints.

2.3 Video prediction

Video prediction has tasks different from the video generation and it is one of the major problems in the field of computer vision. In particular, the method of modeling the domain of videos is not unified, but in the existing research, the next frame is inferred using the recurrent neural networks like LSTM. In addition, a well-known approach is to estimate intermediate features of dynamic motion using optical flow and human bones [15,17]. However, our model architecture is different from other methods because our model does not use optical flow and the recurrent neural network. Also, our model is good at handling stationary images as our model learns without their additional information. The cutting-edge study is MD-GAN [29], which predicts future frames from a stationary image. However, there is a big difference between our model and MD-GAN. The first is that [29] learns in two-stage, but our model learns in one-stage. The second is that MD-GAN [29] leverage the Gram matrix to explicitly model dynamic motion, while our model leverages the spatial self-attention to model by the spatial average weight. In other words, our model is added to the self-spatial attention layer to the first stage structure of MD-GAN.

2.4 Self Attention Mechanism

Recently, there are many works that produced remarkable results using the self-attention mechanism [25,28,31]. The self-attention module [25] calculates the response at the position in the feature map by paying attention not only to the surroundings of an attending point but also to all the positions in the image and taking a weighted average of them. Non-local Neural Networks [28] proposes a non-local operator which handles global information in spatial and temporal directions using the self-attention method [25]. Similarly, our spatial self-attention is likewise based on a self-attention mechanism. However, in the case of frame prediction and generation, because all the frames are equally important, only the spatial direction is used without considering the time direction.

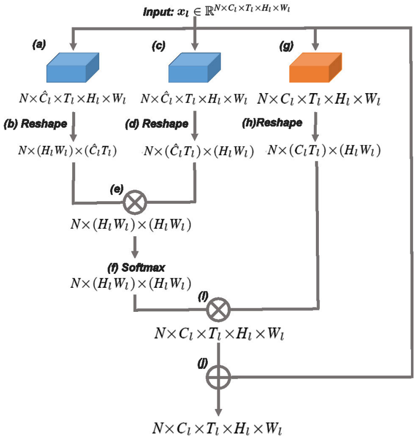
Input: $x_l \in \mathbb{R}^{N\times C_l\times T_l\times H_l\times W_l}$
(a)
(c)
(g)
$N\times\hat{C}_l\times T_l\times H_l\times W_l$
$N\times\hat{C}_l\times T_l\times H_l\times W_l$
$N\times C_l\times T_l\times H_l\times W_l$
(b) Reshape
(d) Reshape
(h)Reshape
$N\times(H_lW_l)\times(\hat{C}_lT_l)$
$N\times(\hat{C}_lT_l)\times(H_lW_l)$
$N\times(C_lT_l)\times(H_lW_l)$
(e)
$N\times(H_lW_l)\times(H_lW_l)$
(f) Softmax
$N\times(H_lW_l)\times(H_lW_l)$
(i)
$N\times C_l\times T_l\times H_l\times W_l$
(j)
$N\times C_l\times T_l\times H_l\times W_l$

Fig. 2. The overview architecture of our spatial self-attention mechanism. The feature maps are shown as the shape of their tensors. "⊗" denotes a matrix multiplication, and "⊕" denotes element-wise sum. The softmax operations are calculated in each column. The blue box changes the matrix of channel size C_l to $\hat{C}_l$ and outputs it, but the orange box outputs a matrix with the channel size C_l.

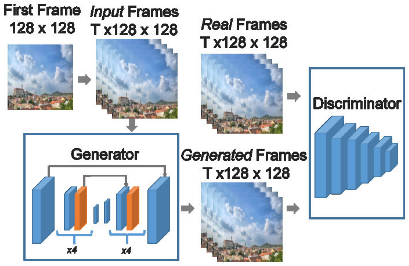

Fig. 3. The overview architecture of our SSA-VGAN. Blue layers indicate 3D convolutional layers and 3D deconvolutional layers, and orange layers indicate the spatial self-attention layers. The generator consists of an architecture like 3D U-Net, preventing skip connection from missing content. The input image is duplicated T times from the first frame of the ground truth.

3 Our Approach

In this section, we first describe our proposed Spatial Self-Attention GAN, a framework to generate video samples in one-stage learning. Next, we describe the spatial self-attention module. Finally, we describe the objective function to learn our model.

3.1 Spatial Self-Attention GAN

As shown in Fig. 3, SSA-GAN is a generative adversarial network composed of the generator G and the discriminator D. G consists of 3D U-Net [4] with the skip connection which prevents future frames from blurring and losing content information. In addition, G has the spatial self-attention module based on self-attention [25] followed by each convolutional and deconvolutional layers. By using the module, it enables G to efficiently learn spatial features. D consists of the same structure as the encoder part of the generator and has a sigmoid function in the final layer except for the spatial self-attention layers. D takes real and fake videos as input and tries to distinguish them. Our model learns to take a single RGB frame and predict next T future frames as realistic as possible by one-stage learning.

3.2 Spatial Self-Attention Module

We propose to introduce a spatial self-attention module to learn the long-range dependence within a frame, which allows the network to first rely on the cues in only neighboring pixels and then gradually learn to assign more weight to areas outside the neighborhood. In other words, it enables the network to learn simple tasks firstly and to gradually increase the complexity of the task to get better features. Each l-th layer of the convolution and deconvolution output is $\boldsymbol{x_l} \in \mathbb{R}^{N\times C_l\times T_l\times H_l\times W_l}$where (N, C_l, T_l, H_l, W_l) are the batch size, the number of channels, length of the time axis, the height and the width of the feature maps, respectively. As shown in Fig. 2, (a) the spatial self-attention layer firstly applies the 3D convolution to the input feature $\boldsymbol{x_l}$ and obtains $\boldsymbol{x_{l_1}} \in \mathbb{R}^{N\times \hat{C}_l\times T_l\times H_l\times W_l}$ and (b) resizes to $\hat{\boldsymbol{x}}_{\boldsymbol{l_1}} \in \mathbb{R}^{N\times(H_l W_l)\times(\hat{C}_l T_l)}$. Next, (c) the layer gets $\boldsymbol{x_{l_2}} \in \mathbb{R}^{N\times \hat{C}_l\times T_l\times H_l\times W_l}$ by (a) the same operation and (d) resizes to $\hat{\boldsymbol{x}}_{\boldsymbol{l_2}} \in \mathbb{R}^{N\times(\hat{C}_l T_l)\times(H_l W_l)}$. Furthermore, (e) after calculating the matrix multiplication of $\hat{\boldsymbol{x}}_{\boldsymbol{l_1}}$ and $\hat{\boldsymbol{x}}_{\boldsymbol{l_2}}$, (f) softmax calculate to obtain the attention $\hat{\boldsymbol{X}}_{\boldsymbol{l}} \in \mathbb{R}^{N\times(H_l W_l)\times(H_l W_l)}$, defined as

$$\hat{\boldsymbol{X}}_{\boldsymbol{l}} = \frac{\exp(\boldsymbol{X_l})}{\sum \exp(\boldsymbol{X_l})}, \text{where } \boldsymbol{X_l} = \boldsymbol{x_{l_1}} \otimes \boldsymbol{x_{l_2}}. \tag{1}$$

This represents the weighted average inside the feature map. Following, (g) the layer applies the 3D convolution to the input feature $\boldsymbol{x_l}$ and obtains $\boldsymbol{x_{l_3}} \in \mathbb{R}^{N\times C_l\times T_l\times H_l\times W_l}$ and (h) resizes to $\hat{\boldsymbol{x}}_{\boldsymbol{l_3}} \in \mathbb{R}^{N\times(C_l T_l)\times(H_l W_l)}$.

Then, (i) the resized output of the layer is $\boldsymbol{o} \in \mathbb{R}^{N\times C_l\times T_l\times H_l\times W_l}$, defined as

$$\boldsymbol{o_l} = \hat{\boldsymbol{X}}_{\boldsymbol{l}} \otimes \boldsymbol{x_{l_3}}. \tag{2}$$

Finally, (j) the layer multiplies the output o_l scale parameter γ and calculates the sum of it with the input feature map x_l. Therefore, the final output is y_l, defined as

$$\boldsymbol{y_l} = \gamma \boldsymbol{o_l} + \boldsymbol{x_l}, \tag{3}$$

where γ is a parameter initialized with 0. We leverage $\hat{C}_l = C_l$ for all experiments. We describe the role of the parameter γ in Sec 5.4.

3.3 Spatial Self-Attention GAN Objectives

Our goal is to predict future frames from the stationary image in the one-stage learning model and to predict it as realistically as possible.

Adversarial Loss. In order to make the generated future frame more realistic, we adopt an adversarial loss

$$\begin{aligned}\mathcal{L}_{adv} = \min_{G} \max_{D} \mathop{\mathbb{E}}_{Y\sim\mathbb{P}_r} [\log D(Y)] + \\ \mathop{\mathbb{E}}_{\bar{X}\sim\mathbb{P}_g} [\log (1 - D(\bar{X}))],\end{aligned} \tag{4}$$

where Y is sampled from the data distribution $\mathbb{P}_r$ and $\bar{X}$ is sampled from the model distribution $\mathbb{P}_g$ implicitly defined by $\bar{X} = G(X), X \ \sim \mathbb{P}_r$. The generator predicts a future frame $\bar{X}$ from a stationary video X to fool the discriminator, while the discriminator tries to distinguish between real and fake frames.

Content Loss. Previous approach [19] indicates that it is more beneficial to combine traditional loss like $L1$ norm and $L2$ norm with the adversarial loss. Although the role of the discriminator remains unchanged, the role of the generator play a role not only to fool the discriminator but also to generate the fake images closer to the real ones. In addition, pix2pix [11] shows that the output images become less blurred at $L1$ norm than $L2$ norm. To ensure that the content of the generated frames is a pattern similar to the content of the real video, the content objective is defined as complementing the adversarial objective,

$$\mathcal{L}_{con} = \mathop{\mathbb{E}}_{Y\sim\mathbb{P}_r, \bar{X}\sim\mathbb{P}_g} [\|Y - \bar{X}\|], \tag{5}$$

where the generator tries to generate a frame similar to Y at the pixel level.

Full Objective. Finally, the loss objectives which optimize the generator and the discriminator are defined as

$$\mathcal{L}_D = -\mathcal{L}_{adv}, \tag{6}$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{con}\mathcal{L}_{con}, \tag{7}$$

where λ_{con} is a hyperparameter that controls the relative importance of content loss compared to the adversarial loss. We leverage $\lambda_{con} = 1$ for all experiments.

4 Implementation Details

As shown in Fig. 3, SSA-GAN is composed of the generator of the 3D U-Net architecture [4] with the skip-connection and the discriminator to prevent future frames from blurring or losing content information. The skip connection is useful as identity mapping [9]. The generator network consists of a six convolution layer, six transposed convolutions, and skip connection. In addition, the

Table 1. The architecture of the generator.

Layer	Filters	Kernel	Stride	Padding
conv1	32	(3, 4, 4)	(1, 2, 2)	(1, 1, 1)
conv2	64	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
conv3	128	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
conv4	256	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
conv5	512	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
conv6	512	(2, 4, 4)	(1, 1, 1)	(0, 0, 0)
deconv1	512	(2, 4, 4)	(1, 1, 1)	(0, 0, 0)
deconv2	256	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
deconv3	128	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
deconv4	64	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
deconv5	32	(4, 4, 4)	(2, 2, 2)	(1, 1, 1)
deconv6	3	(3, 4, 4)	(1, 2, 2)	(1, 1, 1)

generator has the spatial self-attention module following each convolutional and deconvolutional layers. We apply Batch Normalization [10] to all 3D convolutional layers except the first and last layers, followed by Leaky ReLU and ReLU. The output layer exploits Tanh as an activation function for the generator. We adopt Adam as the optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.9$. The learning rate is fixed at 0.0002 during learning. We perform one generator update after five discriminator updates as in [8]. We set the batch size to 16 for all experiments. We used the same architecture as [29] regarding the architecture of the generator network, as shown in Table 1. The architecture of the discriminator is the same as the convolutional parts of the generator.

5 Experiments

In this section, we first describe two data sets used in the experiment. Next, we conduct to compare the models learned in those data sets with related studies.

Finally, we discuss the parameters *gamma* that manipulate our proposed spatial self-attention in Eq 3.

5.1 Datasets

To evaluate the robustness and effectiveness of our approach, we compare our model with other approaches using two datasets, which are the cloud time-lapse dataset [29] and the beach dataset [26].

Cloud Time-Lapse Dataset. We leverage the time lapse video dataset [1] gathered from the Internet [29] for evaluation. The dataset includes over 5,000 time-lapse videos collected from Youtube. The videos are cut into short clips and include those containing dynamic sky scenes such as the cloudy sky with moving clouds, and the starry sky with moving stars. In addition, the dataset consists

[1] https://sites.google.com/site/whluoimperial/mdgan

Table 2. Quantitative comparison results on the cloud time-lapse dataset. The value range of POS is $[0, 1000]$.

”Which is more realistic?”	POS
Prefer Ours over [29] Stage *I*	871
Prefer Ours over [29] Stage *II*	526
Prefer [29] Stage *I* over Real	286
Prefer [29] Stage *II* over Real	322
Prefer Ours over Real	334

Table 3. Experiment results on the cloud time-lapse dataset by MSE, PSNR, and SSIM. Ours (a) and Ours (b) are models which proposed layers are added to the first and second stages of MD-GAN [29], respectively.

Method	MSE↓	PSNR↑	SSIM↑
[29] *I*	0.0280	23.14	0.5997
[29] *II*	0.0245	23.8529	0.6327
Ours (a)	**0.0238**	**24.3512**	**0.6991**
Ours (b)	0.0259	23.5224	0.6460

of 35,392 training video clips and 2,815 testing video clips each containing 32 frames. However, the original size of each frame is $3 \times 640 \times 360$, and we resize it into a square image size $3 \times 128 \times 128$. We duplicate the first frame of the input video 32 times to make it a static input video. We normalized the inputs by converting the color value to $[-1, 1]$.

Beach Dataset. We leverage the unlabeled video dataset which is released by [26] [2], which do not contain any time-lapse video. We divide the dataset of 10% into training data and 90% into evaluation data.

5.2 Experiments on the Cloud Time-Lapse dataset

In this section, we evaluate the performance of SSA-GAN for a quantitative evaluation. As a baseline model, we adopt MD-GAN, which is the method of performing the highest accuracy using the cloud time-lapse dataset. In addition, we also experiment with our model (a) to learn Stage *I* and our model (b) to learn Stage *II* that introduced our proposed layer at each stage of MD-GAN.

To evaluate whether the predicted future frames is more natural, we compare these models in each pair in the same way as [29]. We prepare 100 pairs of videos according to the five cases shown in Table 2, which is selected randomly from the evaluation dataset. We show ten subjects the pairs of generated video and ask them ”which is more realistic?”. Then, we count the answers of their evaluation, which means Preference Opinion Score (POS). The results generated from our model randomly appear in either left or right side in the test to get a more reliable evaluation. As shown in Table 2, our model achieved the better results than other models. We demonstrate that the spatial self-attention module generates dynamic cloud motion prediction from all spatial relationships in the image.

Finally, for each approach, we calculate the Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index (SSIM) between the full of evaluation datasets. As shown in Table 3, our model (a) shows better performance than other methods.

[2] http://www.cs.columbia.edu/~vondrick/tinyvideo/

Table 4. Experiment results on the Beach dataset by MSE, PSNR, and SSIM.

Method	MSE↓	PSNR↑	SSIM↑
RNN-GAN	0.1849	7.7988	0.5143
VGAN	0.0958	11.5586	0.6035
MD-GAN Stage *II*	0.0422	16.1951	**0.8019**
Ours (a)	0.0379	23.6601	0.7320
Ours (b)	**0.0374**	**25.6432**	0.7346

5.3 Experiments on the Beach Dataset

In this section, we compare our model with MD-GAN, VGAN, and RNN-GAN [33] using the beach dataset in a quantitative evaluation. All models generate 32 future frames and are trained using the adversarial loss. VGAN and RNN-GAN take an image of 64 × 64 resolution and predict future frames of 64 × 64 resolution. In addition, MD-GAN takes also the same resolution image to satisfy these conditions. Therefore, for a fair comparison, our model is also adjusted to learning with a 64 × 64 resolution image. To learn this model, our model was removed the first convolutional and deconvolutional layer so that model can predict future frames of resolution 64 × 64. All models calculate MSE, PSNR, and SSIM using randomly sampled 1000 videos from the evaluation dataset. As shown in Table 4, our model showed the better scores than the other models regarding PSNR an MSE, although the MD-GAN Stage *II* achieved the best score in SSIM.

5.4 Discussion

We conduct ablation studies to verify the important role of the parameter *gamma* in Eq 3. The parameter *gamma* is initialized to 0 and is the weight of the spatial self-attention module. Figure 4 shows the difference of the prediction result by (A) existence (B) non existence of gamma parameter for performing qualitative evaluation. In many cases, Method (B) fails to learn motion generation as the red arrows in Figure 4. The issue is because the spatial self-attention module is affected by over-weighting the entire image. Thus, although the generator generates images with quality that can fool the discriminator, it fails to capture the movement of the cloud. Method (A) overcomes this problem by adjusting the influence while learning the overall weight of our module.

6 Conclusion

We propose SSA-GAN with the spatial self-attention mechanism based on the self-attention. The spatial self-attention mechanism enables the models to represent features of real samples. In addition, the mechanism makes it possible to learn models efficiently in the one-stage of end-to-end learning. We demonstrate

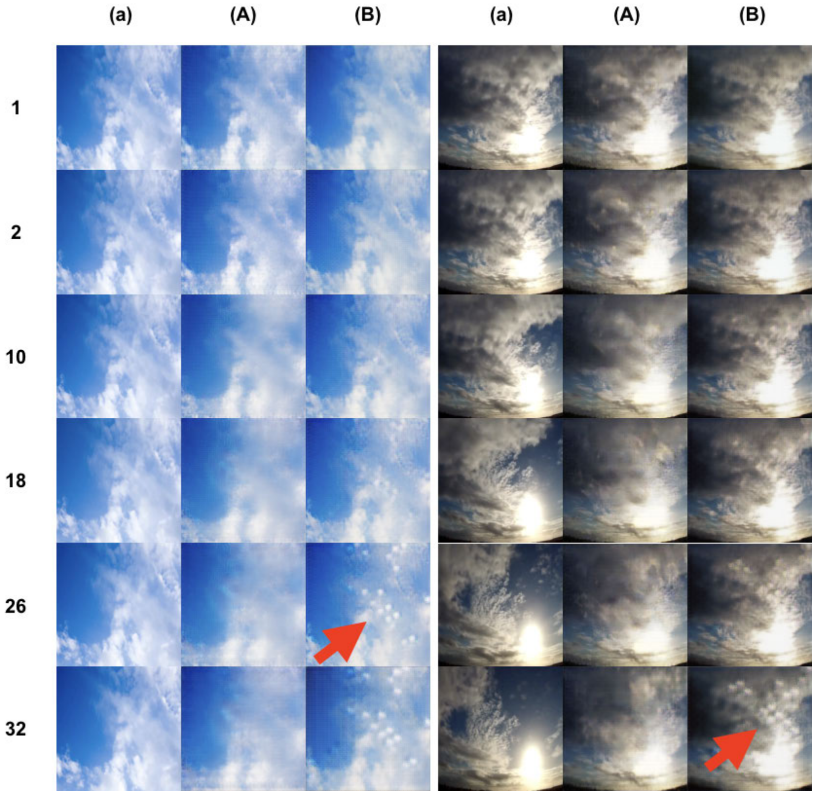

Fig. 4. Some example results of cloud motions generated by our proposed model. From left to right: (a) the ground truth, (A) existence, (B) non existence of *gamma* parameter in Eq 3, respectively. The red arrow shows an example of failing to capture the movement of the cloud.

that our model achieves comparable results with the state-of-the-art method which performs a two-stage learning and predict future frames as realistically as possible.

However, our proposed model fails to generate cloud motion as much as ground truth video. In the future, we use temporal features to learn the model that can reproduce time-lapse video. More experimental results including generated videos can be seen at https://udonda.github.io/ACPR_44_supplementary_material/.

Acknowledgements: This work was supported by JSPS KAKENHI Grant Number 15H05915, 17H01745, 17H06100 and 19H04929.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: International Conference on Machine Learning(ICML) (2017)
2. Cai, H., Bai, C., Tai, Y., Tang, C.: Deep video generation, prediction and completion of human action sequences. In: Proc.of European Conference on Computer Vision(ECCV) (2018)
3. Cheng, X., Dale, C., Liu, J.: Understanding the characteristics of internet short video sharing: Youtube as a case study. In: Multimedia (ISM) 2012 IEEE International Symposium (2007)
4. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: Learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing & Computer Assisted Intervention(MICCAI) (2016)
5. Finn, C., Goodfellow, I., Levine, S.: Unsupervised learning for physical interaction through video prediction. In: Proc.of Neural Information Processing Systems (2016)
6. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proc.of Neural Information Processing Systems (2014)
7. Gu, C., Sun, C., Vijayanarasimhan, S., Pantofaru, C., Ross, D.A., Toderici, G., Li, Y., Ricco, S., Sukthankar, R., Schmid, C., Malik, J.: AVA: A video dataset of spatio-temporally localized atomic visual actions. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2018)
8. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Proc.of Neural Information Processing Systems (2018)
9. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Proc.of European Conference on Computer Vision(ECCV) (2016)
10. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning(ICML) (2015)
11. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2017)
12. Jhuang, H., Gall, J., Zuffi, S., Schmid, C., Black, M.J.: Towards understanding action recognition. In: Proc.of IEEE International Conference on Computer Vision(ICCV) (2013)
13. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. In: "Proc. of the International Conference on Learning Representation(ICLR)" (2017)
14. Krishna, R., Hata, K., Ren, F., Li, F., Niebles, J.C.: Dense-captioning events in videos. In: Proc.of IEEE International Conference on Computer Vision(ICCV) (2017)
15. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.: Flow-grounded spatial-temporal video prediction from still images. In: Proc.of European Conference on Computer Vision(ECCV) (2018)
16. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. In: Proc. of the International Conference on Learning Representation(ICLR) (2016)

17. Ohnishi, K., Yamamoto, S., Ushiku, Y., Harada, T.: Hierarchical video generation from orthogonal information: Optical flow and texture. In: Proc.of AAAI Conference on Artificial Intelligence(AAAI) (2018)
18. Pan, P., Xu, Z., Yang, Y., Wu, F., Zhuang, Y.: Hierarchical recurrent neural encoder for video representation with application to captioning. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2016)
19. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2016)
20. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: Proc. of the International Conference on Learning Representation(ICLR) (2016)
21. Saito, M., Matsumoto, E.: Temporal generative adversarial nets. In: Proc.of IEEE International Conference on Computer Vision(ICCV) (2017)
22. Shou, Z., Wang, D., Chang, S.: Action temporal localization in untrimmed videos via multi-stage cnns. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2016)
23. Srivastava, N., Mansimov, E., Salakhutdinov, R.: Unsupervised learning of video representations using lstms. In: International Conference on Machine Learning(ICML) (2015)
24. Tulyakov, S., Liu, M., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2018)
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Proc.of Neural Information Processing Systems (2017)
26. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: Proc.of Neural Information Processing Systems (2016)
27. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2013)
28. Wang, X., Girshick, R.B., Gupta, A., He, K.: Non-local neural networks. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2018)
29. Xiong, W., Luo, W., Ma, L., Liu, W., Luo, J.: Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks. In: Proc.of IEEE Computer Vision and Pattern Recognition(CVPR) (2018)
30. Yang, C., Wang, Z., Zhu, X., Huang, C., Shi, J., Lin, D.: Pose guided human video generation. In: Proc.of European Conference on Computer Vision(ECCV) (2018)
31. Zhang, H., Goodfellow, I.J., Metaxas, D.N., Odena, A.: Self-attention generative adversarial networks. arXiv:1805.08318 (2018)
32. Zhao, L., Peng, X., Tian, Y., Kapadia, M., Metaxas, D.N.: Learning to forecast and refine residual motion for image-to-video generation. In: Proc.of European Conference on Computer Vision(ECCV) (2018)
33. Zhou, Y., Berg, T.L.: Learning temporal transformations from time-lapse videos. In: Proc.of European Conference on Computer Vision(ECCV) (2016)
34. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networkss. In: Proc.of IEEE International Conference on Computer Vision(ICCV) (2017)