Video Prediction for Precipitation Nowcasting*

Yuan Cao[1], Qiuying Li[1], Lei Chen[2], Junping Zhang[1], and Leiming Ma[2]

[1] Shanghai Key Laboratory of Intelligent Information Processing,
School of Computer Science, Fudan University, Shanghai 200433, China
{caoy16,18210240116,jpzhang}@fudan.edu.cn
[2] Shanghai Observatory, Shanghai, China
qqydss@163.com, malm@typhoon.org.cn

Abstract. Video prediction, which aims to synthesize new consecutive frames subsequent to an existing video. However, its performance suffers from uncertainty of the future. As a potential weather application for video prediction, short time precipitation nowcasting is a more challenging task than other ones as its uncertainty is highly influenced by temperature, atmospheric, wind, humidity and such like. To address this issue, we propose a star-bridge neural network (StarBriNet). Specifically, we first construct a simple yet effective star-shape information bridge for RNN to transfer features across time-steps. We also propose a novel loss function designed for precipitaion nowcasting task. Furthermore, we utilize group normalization to refine the predictive performance of our network. Experiments in a Moving-Digital dataset and a weather predicting dataset demonstrate that our model outperforms the state-of-the-art algorithms for video prediction and precipitation nowcasting, achieving satisfied weather forecasting performance.

Keywords: Video Prediction · Percipitation Nowcasting · Video Prediction · ConvLSTM.

1 Introduction

Video prediction, which aims to synthesize new consecutive frames subsequent to an existing video, has broad applications in autonomous driving, task planning, weather forecasting, and new view synthesis. However, its performance suffers from uncertainty of the future. As a high challenging task in video prediction, Nowcasting Convective Precipitation (NCP) aims to forecast precise rainfall intensity for a specified region over a six-hour period. Application of NCP has a significant importance to agriculture, flood alerting, daily life of citizens and transportation. A commonly-used technique for NCP is to perform extrapolation based on radar echoes, whose history can be dated back to the early 1960s. [12] However, traditional Numerical Weather Prediction(NWP) models easily suffer from the imprecision of original data when their measurement is low-precision. Consequently, these models may have non-robust performance, especially in the

* Supported by organization Fudan University.

first few fours. As the increase of huge amount of video-like radar echo data, fortunately, there exists high opportunities to refine the performance of NCP by utilizing machine learning and video prediction-related methods. For example, [5] decomposed the input video frames into poses and contents to predict each content separately, and then predicted better future images. [7,1,2] estimated multiple high quality possible future frames by sample on a latent feature space learned from a variational auto-encoder.

With the consecutive 6-minute-interval radar echo frames as the input and output data, [10,11] dealt the issue of NCP based on the framework of video prediction. Specifically, [10] proposed encoder-decoder based convolutional LSTM, which is broadly used in various video prediction tasks. Predictive RNN (PredRNN)[15] and PredRNN++[14] achieved the state-of-the-art by combining ConvLSTM with a new Spatial-Temporal-LSTM structure, and applying layer normalization together with a zigzag connection across different LSTM layers. Although video prediction methods can be applicable to the NCP task, the performance of their models suffers from stochastic and unpredictable existed in the real-world events[1]. The reason is that NCP is highly influenced by temperature, atmosphere, wind, humidity and such like. Since all of these meteorological parameters cannot be captured by radar echo images, it brings new challenges to video prediction methods.

In this paper, we first propose a simple yet effective star-shape information bridge to transfer features across time-steps. We also discuss about group normalization which significantly improves the performance of our network. Inspired by Critical Success Index (CSI) skill score from weather forecasting field, we propose a novel multi-sigmoid loss for NCP task. We evaluate our StarBriNet on two datasets: a Moving MNIST dataset and a Radar Echo dataset of east China. Experiments demonstrate that we achieve the state-of-the-art on both datasets.

The contributions of this paper are stated as follows:

1. We propose an efficient end-to-end network SBLSTM for precipitation convective nowcasting.
2. We propose an effective star-shape infomation bridge for recurrent network.
3. We propose a new multi sigmoid loss for precipitation nowcasting task.

2 Related Works

In this section, we will briefly survey the development of convolutional LSTM, precipitation nowcasting problem and group normalization.

2.1 Convolutional LSTM.

Video prediction has received increasing attention in the recent years. Generally speaking, the nontrivial task of video prediction has been studied based on Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). Methods based on CNNs are well-known for extracting features from images

and dealing with complicated images and videos[3]. However, using pure convolutional networks is limited by an inability to deal with temporal information from input video thus, creating a crucial problem in video prediction. Apart from CNN-based methods, researchers also dig into RNN-based methods for video prediction. Convolutional LSTM[10] is an effective one within various RNN methods. Following the Long Short Term Memory model, ConvLSTM utilizes its recurrent neural network architecture to memorize temporal information in a video sequence and extracts the spatial feature maps by using convolutional operation. Therefore, we refine it for better prediction performance in this paper.

2.2 Precipitation Nowcasting.

The task of video prediction is of great value to precipitation nowcasting problem. Previous works such as [10] and [15] have studied in video prediction and applied their models to realistic precipitation forecasting, like Figure 1.

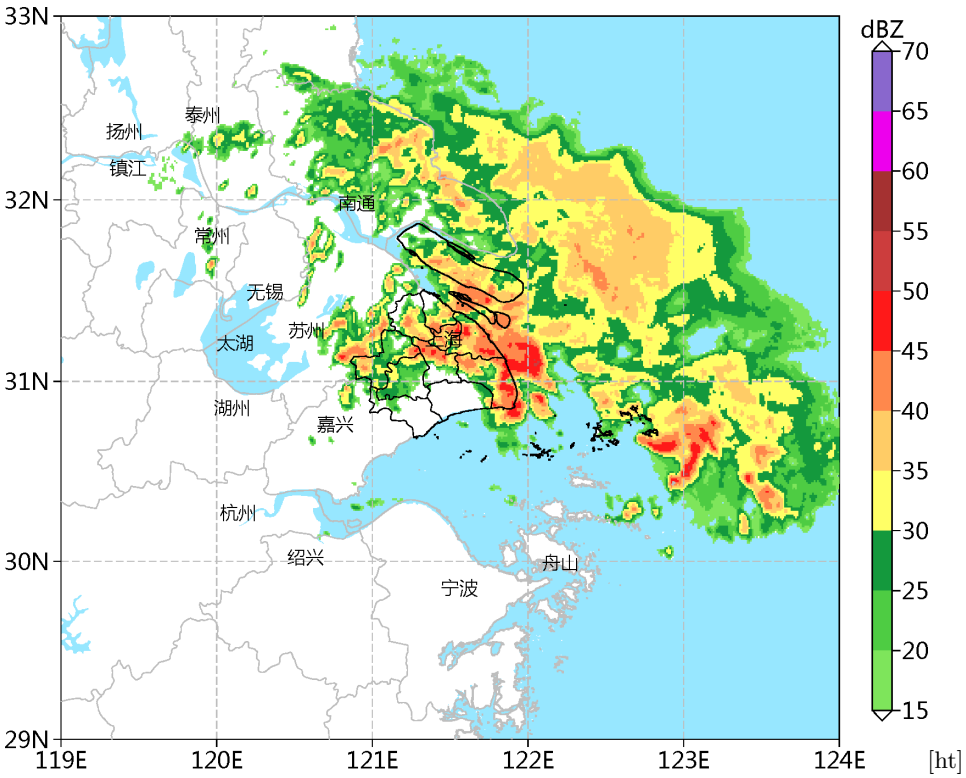

Fig. 1. A precipitation image from Radar Echo Dataset of east China. We plot the image with pseudo-color and map to the with east China map.

Given a sequence of input frames, e.g. sequence of radar echoes, our goal is to predict the most possible sequence of future frames. In paper [10], ConvLSTM follows Equation 1 to make prediction on Moving MNIST dataset and Hong Kong radar echo dataset. PredRNN enhances the ConvLSTM by using a spatiotemporal flow to further utilize the spatial feature and memorize the temporal information. Following ConvLSTM, PredRNN trains on Moving MNIST dataset and also deals with the precipitation nowcasting problem by training on the HK radar echo data.

2.3 Group Normalization.

As we all know, the popular Batch Normalization (BN) plays an important role in training deep neural networks. BN computes the mean and variance of a set of mini batch to normalize feature maps. Nevertheless, BN is restricted when being applied to small batch size problems. When it comes to deal with high resolution images or even complicated video data, small batches are inevitable hence BN always shows limited performance.

Recently, [16] proposed a novel form of normalization named Group Normalization (GN). Distinct from the widespread BN, GN normalizes feature maps by grouping channels into small groups and computing the mean and variance of each small group. In this way, GN can normalize the feature tensors within each single sample. As a result, the normalization can be independent from batch size. In the setting of small batches which is commonly seen in computer vision tasks, GN can be an effective normalization without involving the batch dimension.

We discover the effectiveness of applying GN as our normalization layer. As video prediction becomes popular with computer vision researchers, we find that no former research has connected the promising GN to video prediction tasks. We will illustrate the distinct effects of using GN in our model and demonstrate how this method makes a difference by showing some comparison results in experiment section.

3 Methods

In this section, we will detail our proposed model. Define a length-T video sequence as $I_{1:T} = [i_1, i_2, ...i_T]$. Each $i_t \in \mathbf{R}^{H \times W}$ stands for a width W, height H gray image, where $t \in \{1, ..., T\}$. Our goal is to predict the next length-L ($L > 1$) sequence of video frames subsequent to $I_{1:T}$. The goal can be formulated as follows:

$$\hat{I}_{T+1:T+L} = \underset{I_{T+1:T+L}}{\arg\max}\, p(I_{T+1:T+L} | I_{1:T}) \tag{1}$$

in which $\hat{I}_{T+1:T+L}$ is the predicted length-L frames.

We present an effective network, Star-Bridge Convolutional LSTM (StarBriNet)[3] network, for the task of video prediction. Specifically, we include an

[3] Code for SBNet are available at `https://github.com/caotong0/SBNet-for-video-prediction`

encoder-decoder ConvLSTM network. And we enhance the origin encoder-decoder architecture by fusing the hidden state of each layer of decoder at $t-1$, followed by generating and distributing these states. We add a novel Star-Shape information bridge in our decoders to extract the spatial information and boost the memorizing ability of temporal information. Figure 2 illustrates our entire model architecture.

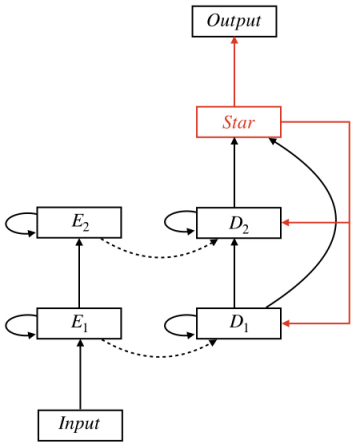

Fig. 2. The entire model.

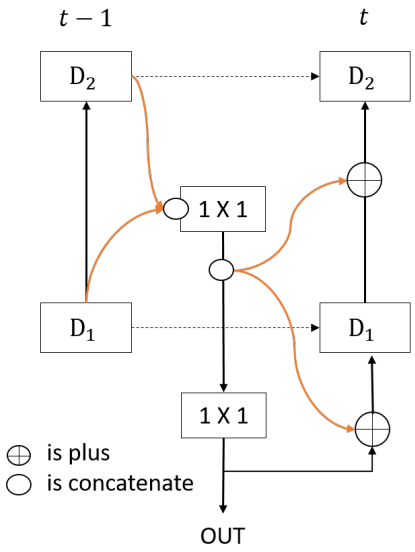

Fig. 3. The Star-Shape flow in Decoder network.

3.1 Star-shape Information Bridge

To start with our task of video prediction, we first utilize the Convolutional LSTM (ConvLSTM) network as our basic building block. The key equations of ConvLSTM are summarized as follows:

$$\begin{aligned} f_t, i_t &= \sigma_g(W_g * [x_t, h_{t-1}, c_{t-1}] + b_g) \\ c_t &= f_t \circ c_{t-1} + i_t \circ tanh(W_c * [x_t, h_{t-1}] + b_c) \\ o_t &= \sigma_o(W_o * [x_t, h_{t-1}, c_t] + b_o) \\ h_t &= o_t \circ tanh(c_t) \end{aligned} \tag{2}$$

Traditional multi-layer LSTMs usually take the last output $\hat{I}_{t-1}$ as the input of the first layer at next time-step. A disadvantage of this zigzag connection is that it may bring accumulative errors to predictions in testing stage, because $\hat{I}_{t-1}$

is different from the ground truth input I_{t-1}. But this connection passes more information across time steps which benefits the back-propagation, and this is the main reason why researcher still use it. Inspired by this, we a novel Star-Shape information bridge to add more information from the last time-step to make the feature flow in multi-layer ConvLSTM more robust. More specifically, we concatenate output of all ConvLSTM layers and pass it to a 1×1 convolution layer and split the output of convolution layer to all ConvLSTM layers of next time-step by a residual connection to their inputs(Figure 3). However, it is hard to train the star-shape structure because of gradient exploding problem. Therefore, we attach the group normalization after each convolution layer and greatly relieve this hard-training problem. We also add group normalization layer to every convolution layer in ConvLSTM which directly improve our prediction performance by a large margin. We will discuss about this boost in next section.

3.2 Group Normalization for Video Prediction

By investigating the research of different normalization techniques, we discover that the normalization layer plays a key role in refining the performance to video prediction. As illustrated in paper [16], the group normalization shows its remarkable effectiveness in small batch size experiments and to a great extent alleviates the problem when using batch normalization and small batches. We experimentally make comparisons between different normalization techniques, and find that the group normalization drastically boosts the performance of our model.

Recurrent neural networks suffer from gradient vanishing and exploding problem since they have this shape $It = W * I_{t-1}$ typically as mentioned in [4]. Therefore LSTM was brought to alleviate this problem by using sigmoid and tanh gate functions to limit the parameters in a range from 0 to 1. Therefore there is no exploding problem left in LSTMs. But then it raises another problem on whether these gates function are the optimal way to solve the gradient issues or not. Meanwhile, CNNs also have to face this problem when the layers go deeper. [3,9] used residual blocks and skip connections to propagate the gradient from a long distance. These tricks successfully stop gradient from vanishing.

On the other hand, [6] is a milestone technique to deal with the problem of gradient vanishing and exploding in deep learning. [8,13,16] enlarged the boundary of normalization, and they launched normalization of parameters to deep networks in small batch size circumstances. We adopt the group normalization, which improves our performance significantly, in our network. In [16], the input channels are separated into g groups, each of which contains c channels. The mean and standard-deviation are calculated separately over the each group.

Group Normalization performs the following computation:

$$
\begin{aligned}
&y_i = \gamma \hat{x}_i + \beta,\ \hat{x}_i = \frac{1}{\sigma_i}(x_i - \mu_i) \\
&\mu_i = \frac{1}{c}\sum_{k \in S_i} x_k,\ \sigma_i = \sqrt{\frac{1}{c}\sum_{k \in S_i}(x_k - \mu_i)^2 + \epsilon} \\
&S_i = \{k | k_N = i_n, \lfloor \frac{k_C}{G} \rfloor = \lfloor \frac{i_C}{G} \rfloor\}
\end{aligned}
\tag{3}
$$

where x is the feature from last layer, and i is an index, γ and β are trainable scale and shift. Here C is the number of channels per group, we set C to 16 for all convolution layers in this work.

3.3 Multi-Sigmoid Loss for Precipitation Nowcasting.

We propose a novel Multi-Sigmoid Loss for NCP task inspired the Critical Success Index (CSI) from area of precipitation nowcasting. The CSI skill score is defined as $CSI = \frac{TP}{TP+FN+FP}$, where TP denotes the number of True Positives, FN denote the number of False Negatives, FP denote the number of False Negatives in Table 1. We could not use CSI as our loss function cause

Table 1. The 20dBZ CSI skill Score

	$R_{GT} > 20dBZ$	$R_{GT} < 20dBZ$
$R_{Pred} > 20dBZ$	True Positive	False Positive
$R_{Pred} < 20dBZ$	False Negative	True Negative

it is not differentiable. So we propose a sigmoid loss L_i^{SSL} to simulate CSI scores at classification point c_i. In this work, we use weather forecasters' recommendation, $\{10, 20, 30, 40\}$ in radar echo scale, as our classification points $\{c_1, c_2, \cdots, c_n\}$. The multi-sigmoid loss is composed of a set of single sigmoid losses $\{L_i^{SSL}\}_{i=1,2,\cdots,n}$, in which L_i^{SSL} is to evaluate if I gives out the correct classification for the classification point c_i:

$$
L_i^{SSL} = |\operatorname{Sigmoid}((I - c_i) * s) - \operatorname{Sigmoid}((\hat{I} - c_i) * s)|_{abs} \tag{4}
$$

where s is the scale factor, a hyper-parameter to control the slope of the sigmoid function, and the superscript SSL is short for single sigmoid loss. Further, the multi-sigmoid loss is defined as:

$$
L^{MSL} = \sum_{i=1}^{n} L_i^{SSL} \tag{5}
$$

where the superscript MSL is short for multi-sigmoid loss. Multi-sigmoid loss outperform $L1 + L2$ loss on NCP task. We will discuss the influence of s in section 4.3.

4 Experiments

In this section, we will evaluate the performance of our proposed model in two video prediction datasets, i.e., a synthesize moving-MNIST digital dataset and a practical east-China radar echos-based precipitation dataset. We also compare our model with several recently published video prediction algorithms.

4.1 Parameter Settings

Our task is to predict the sequence of consecutive future frames subsequent to a given sequence of frames from a video. In these two datasets, the length of both the input sequence and output sequence is 10.

All our experiments are implemented in PyTorch and conducted on 4 NVIDIA GTX 1080Ti GPUs. We train our model with ADAM optimizer and the learning rate is 0.002. The batchsize is setting to 32×4 and we stop training after 23,000 iterations.

4.2 Moving MNIST Dataset

Implementation. For Moving MNIST synthesis dataset, we employ the same settings as [5] [4]. The Moving MNIST is a dataset consisting of multiple moving digits in a 64×64 frame. During training every sequence is generated by sampling MNIST digits and generating trajectories with random velocity and angle. For testing each sequence is from a fixed dataset consisting of 10,000 sequences. Both training data and testing data are sequences of 20 frames. Within each sequence we utilize 10 frames as our input and other 10 as the ground truth frames without overlapping. We train our model on the 2-digit Moving MNIST as well as the 3-digit dataset. The 2-digit and 3-digit comparison results are illustrated in Tables 2 and 3, respectively.

To evaluate the performance of our model, we use the mean square error(MSE) and the binary cross-entropy(BCE) as our evaluation criteria on the 2-digit Moving MNIST dataset. The MSE we use can be defined as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(I_i - \hat{I}_i)^2. \tag{6}$$

We also utilize the MSE and the mean absolute error(MAE) as our evaluation criterion on the 3-digit Moving MNIST.

Results. We train our model on the 2-digit Moving MNIST dataset and compare our model with several representative models in video prediction domain. As shown in Table 2, our model evidently outperforms other state-of-the-art video

[4] Code of DDPAE and Moving MNIST https://github.com/jthsieh/DDPAE-video-prediction

Table 2. Comparison results on 2-digit Moving MNIST

Model	MSE	BCE
FC-LSTM[4]	118.3	341.2
ConvLSTM[]	103.3	367.2
PredRNN[15]	56.8	-
PredRNN++[14]	46.5	-
DDPAE[5]	38.9	223.0
Ours w/o GN	61.4	278.0
Ours with BN	77.2	186.0
Ours w/o Star Flow	31.0	188.0
Ours	**29.1**	**182.0**

Table 3. Comparison results on 3-digit Moving MNIST

Model	MSE	MAE
FC-LSTM[4]	162.4	310.6
ConvLSTM[10]	142.1	281.5
TRAJGRU [11]	134.0	259.7
PredRNN++[14]	81.7	190.8
Ours	**45.0**	**119.5**

prediction models. Specifically, our method significantly surpasses the performance of our baseline, ConvLSTM. Compared to PredRNN++ and DDPAE, meanwhile, our model makes a notable progress in improving the MSE metric to 29.1 per frame. Furthermore, our best model achieve the best BCE down to 182.0.

We also perform several ablation studies on our model. We train our model without the use of group normalization or any other normalization. Experiments indicate that Ours w/o Group Normalization model obtain the MSE of 61.4 and the BCE of 278.0. This means that our method of using group normalization is crucial and nontrivial. We further analyze our model without the Star-Shape flow and gain the 31.0 MSE. This ablation study shows that the Star-Shape flow can also be helpful in improving the performance of our model.

In order to analyze the generalization of our best model, we further train our model on the 3 moving digits MNIST dataset. We report the MSE metric and the MAE metric in Table 3. Comparison with the baseline model ConvLSTM on the 3-digit MNIST dataset experimentally justifies that our model(MSE=45.0) outperforms the baseline(MSE=142.1). The best MSE model, i.e., Ours in Table 3, also achieve the best MAE result as shown in Table 3, which indicates the ability of our model handling multiple objects(e.g. 2 or more digits) in a video.

Apart from the quantitative analysis of our model on Moving MNIST, we present the qualitative analysis in Figure 4. We have 8 columns of generated images and ground truth images. Each column contains 4 pair of images starting with the first ground truth and the corresponding generated image. The first line shows that our model predicts sharp images at the beginning of time step t_0. As the time-step goes, our model keeps the sharpness of images. Especially for the second pair where the 2 digits overlap, our model makes the digits separated and keeps them crisp and sharp. The ability to keep sharp prediction is shown better on the 3-digit Moving MNIST(Figure 5). For the last pair in Figure 5, the digits are firstly entangled together. Our model disentangles them into independent and sharp digits as is shown in the last line(time step t_7).

Fig. 4. 2-digit Moving MNIST data.

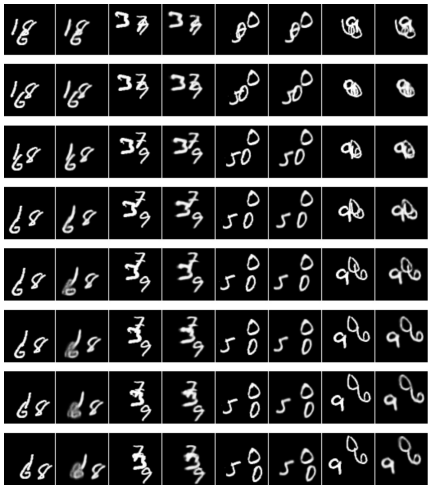

Fig. 5. 3-digit Moving MNIST data.

4.3 East China Radar Echo Dataset

Implementation. Our radar echo dataset contains 170,000 weather radar intensity frames of east China collected from Oct 2015 to Jul 2018 with the minimal interval of 6 minutes. The nowcasting convection precipitation (NCP) problem aims to nowcasting precipitation. Therefore, we filter out rain-less frames and apply a stride-1 sliding window to generate 76779 length-20 sequence samples.

The training set (Oct.2015 to Jul.2017) and testing set (Oct.2017 to Jul.2018) contains 44000 and 32700 sequences samples respectively. For saving training time, we resize the original 501 $\times$ 501 frame to 100 $\times$ 100. We map the echo intensity values R ($0 \leq R \leq 70$) to gray-scale P by $P = R \times \frac{255}{70}$. Compared with Moving MNIST, the NCP task of radar echo data is more tricky and complicate because the moving speed, density and contour of 'rain clouds' on radar echo images are non-constant. To alleviate the uncertainty to some extent, we thus divide the radar echo data into 3 classes based on the mean and first-order derivative of the rain fall intensity. Note that although the cutting strategy may not be optimal, it works in our experiments.

Results. To evaluate the performance of our model, we use the meteorology quantitative metrics CSI (defined in Section 3.3) as well as MSE to evaluate the predicted radar echo frames. CSI stands for a classification accuracy of predictions. We implement PredRNN and ConvLSTM for contrast.

We further demonstrate the qualitative analysis on East-China radar echo dataset. The generated radar echoes and corresponding ground truth are presented in Figure 6.

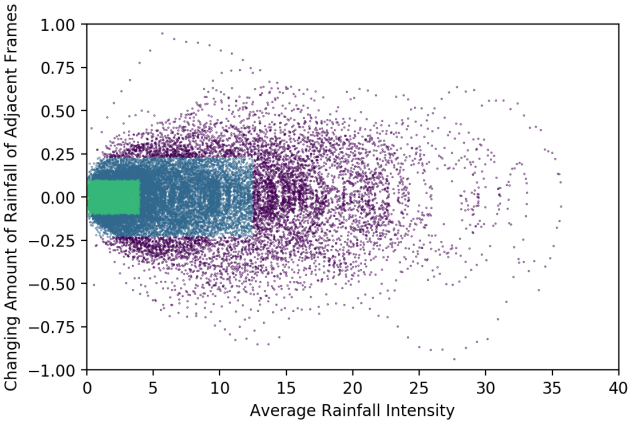

Fig. 6. The Distribution of samples in Radar Echo Dataset. 59.2% samples fall in the little green box, 25.4% for the blue one, and the other 15.4% samples are purple.

Table 4. Comparison results with other models on East-China radar echo dataset. Frozen Prediction uses the last input frame as next 10 time-step predictions directly. Here GN is group normalization, Star is star shape information bridge defined in section 3.2 and MSLoss is multi sigmoid loss in section 3.3

Model	MSE/frame	CSI
Frozen Prediction	10.3	42.1%
ConvLSTM[10]	6.31	59.9%
PredRNN[15]	7.03	63.1%
ConvLSTM+GN	6.51	62.1%
ConvLSTM+GN+Star	**6.14**	64.0%
ConvLSTM+GN+Star+MSLoss	6.18	**64.4%**

Multi-Sigmoid Loss. We carry out experiments to test the influence of the scale factor of Multi-Sigmoid Loss. We run a test for every 350 training iterations and plot them on Figure 7. As shown in the figure, scale-5 and scale-15 outperforms other loss functions including L1+L2. In our opinion, the distribution of precipitation intensities could be better segmented by 10 dBZ, 20 dBZ, 30 dBZ and 40 dBZ CSI accuracies, and on the other hand, professional weather forecasters prefer to use these skill scores to evaluate the performance of predictions in reality.

Figure 6 demonstrates the distribution of the whole dataset, where each point in this figure represents for a length-20 sample sequence of radar echo frames. Y-axis is the changing rate of rain intensities in each sample sequence. Positive value means the rain increased and negative means the rain is letting up. X-axis is the average rainfall intensity of each sample.

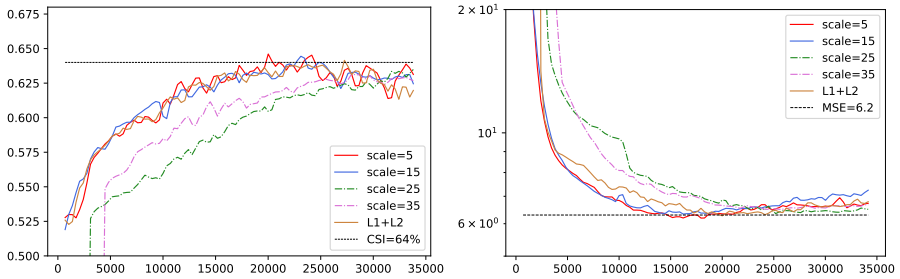

Fig. 7. Evaluation performances of different scale-factors of Multi-Sigmoid loss functions 4 on test dataset of 20dBZ-CSI accuracy and MSE. The left part stands for CSI and the right part stands for MSE. We also plot L1+L2-Loss performance for comparison. X-axis is the number of training iteration.

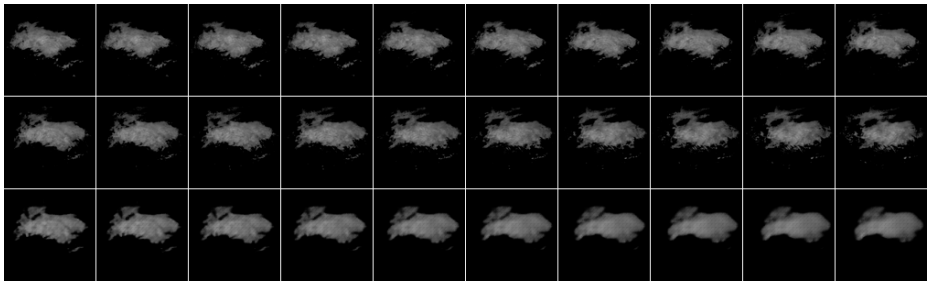

Fig. 8. Predictions on east-China radar echo dataset with a interval of every 6 minutes. From the top to bottom is the Input, Ground Truth and Prediction.

5 Conclusion

We presented Star Shape Information Bridge Network(StrBriNet), a video prediction model that effectively passes feature and gradient through RNN layers. We proposed a new multi-sigmoid loss function for precipitation nowcasting. We performed quantitative analysis on a Moving MNIST dataset and a Radar Echo dataset of east China. The results on both dataset demonstrated that compared with other method, our model achieves the state-of-the-art.

References

1. Babaeizadeh, M., Finn, C., Erhan, D., Campbell, R.H., Levine, S.: Stochastic variational video prediction (2017)
2. Denton, E., Fergus, R.: Stochastic video generation with a learned prior. In: International Conference on Machine Learning. pp. 1182–1191 (2018)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90

4. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
5. Hsieh, J.T., Liu, B., Huang, D.A., Fei-Fei, L.F., Niebles, J.C.: Learning to decompose and disentangle representations for video prediction. In: Advances in Neural Information Processing Systems. pp. 517–526 (2018)
6. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating deep network training by reducing internal covariate shift. CoRR **abs/1502.03167** (2015)
7. Lee, A.X., Zhang, R., Ebert, F., Abbeel, P., Finn, C., Levine, S.: Stochastic adversarial video prediction. arXiv preprint arXiv:1804.01523 (2018)
8. Lei Ba, J., Kiros, J.R., Hinton, G.E.: Layer Normalization. arXiv e-prints (Jul 2016)
9. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 234–241 (2015)
10. Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W., Woo, W.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Proceedings of the International Conference on Neural Information Processing Systems. pp. 802–810 (2015)
11. Shi, X., Gao, Z., Lausen, L., Wang, H., Yeung, D.Y., Wong, W.k., WOO, W.c.: Deep learning for precipitation nowcasting: A benchmark and a new model. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 5617–5627. Curran Associates, Inc. (2017)
12. Sun, J., Xue, M., Wilson, J.W., Zawadzki, I., Ballard, S.P., Onvleehooimeyer, J., Joe, P., Barker, D.M., Li, P.W., Golding, B.: Use of nwp for nowcasting convective precipitation: recent progress and challenges. Bulletin of the American Meteorological Society **95**(95), 409–426 (2014)
13. Ulyanov, D., Vedaldi, A., Lempitsky, V.S.: Instance normalization: The missing ingredient for fast stylization. CoRR **abs/1607.08022** (2016), `http://arxiv.org/abs/1607.08022`
14. Wang, Y., Gao, Z., Long, M., Wang, J., Yu, P.S.: Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning (2018)
15. Wang, Y., Long, M., Wang, J., Gao, Z., Yu, P.S.: Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 879–888. Curran Associates, Inc. (2017)
16. Wu, Y., He, K.: Group normalization. In: The European Conference on Computer Vision (ECCV) (September 2018)