ASTM: AN ATTENTION BASED SPATIOTEMPORAL MODEL FOR VIDEO PREDICTION USING 3D CONVOLUTIONAL NEURAL NETWORKS

Zheng Chang[*], *Xinfeng Zhang*[*], *Shanshe Wang*[†‡], *Siwei Ma*[†‡], *Yan Ye*[⋆] *and Wen Gao*[†]

[*]University of Chinese Academy of Sciences, Beijing, China
[†]Institute of Digital Media, Peking University, Beijing, China
[‡]Information Technology R&D Innovation Center of Peking University, Shaoxing, China
[⋆]Alibaba Group, Beijing, China
Email: changzheng18@mails.ucas.ac.cn, xfzhang@ucas.ac.cn,
{sswang, swma, wgao}@pku.edu.cn, yan.ye@alibaba-inc.com

2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428362

ABSTRACT

Video prediction has always been a challenging task in video representation learning due to the diversity of spatial-temporal evolution in videos. In this paper, we propose an Attention based SpatioTemporal Model for Video Prediction based on 3D Convolutional Neural Networks and Long Short-Term Memory (LSTM), which is named *ASTM*. In our method, we leverage both multi-term and short-term inter-frame dependencies in temporal domain to capture reliable motion information for videos. In particular, we design an *Efficient Inter-Frame Attention Gate* (EIFAG) to efficiently aggregate the multi-term inter-frame dependencies and integrate 3D convolutional operations into the proposed model to further improve the local perception to videos by capturing more accurate short-term temporal dependency. In addition, we make use of the multi-layer Spatiotemporal LSTM (PredRNN) structure to preserve more spatial appearance details for videos. To evaluate the adaptability of our model on more complex real scenes, we collect a multi-level spatiotemporal (MLST) dataset. Experimental results show that the proposed model can achieve state-of-the-art performance on both widely used datasets and the proposed MLST dataset.

Index Terms— Recurrent neural network, 3D convolutional neural network, sequence learning, attention, long short-term memory, video prediction.

1. INTRODUCTION

Video Prediction plays an important role in many video processing applications such as video coding [1, 2], video classification [3] and so on. However, learning a promising representation for videos is a very challenging task in machine learning and computer vision tasks, for the fact that we need to consider both the complex spatial frame appearance information and the time-varying temporal frame dependency information in videos.

In recent decades, deep learning technologies have shown their great power in learning efficient features in many domains, especially in computer vision and natural language processing Motivated by this, recently, many learning-based approaches have been proposed to solve the video prediction problem, which can be roughly summarized into two categories: convolutional neural network (CNN) based methods and recurrent neural network (RNN) based methods.

Most CNN-based methods for video prediction [4–7] only utilize CNNs to learn spatial features for video inputs, and output the next frame based on the learned spatial features. Although CNNs have achieved satisfactory performance in next-frame prediction task, they may not have enough ability to extract promising temporal features for the whole video and the performance in multi-frame prediction task is needed to be improved. In addition, as the period of the predicted videos increasing, the computation load increases dramatically, which is unacceptable. To solve the above two problems in CNN-based methods, RNN-based methods have been proposed for video prediction for their advantages in precessing sequence data and low computation load. In particular, RNN with the Long Short-Term Memory architecture [8] (LSTM) may be a potential solution to video prediction problem with more promising performance due to its great power for capturing long short-term temporal dependencies.

LSTMs have been used successfully to conduct a variety of sequence learning tasks, such as machine translation [9], speech recognition [10] and so on. Since videos can be treated as a special kind of sequential data, in recent studies, LSTMs have been applied on video prediction problem. Srivastava *et al.* [11] extended the LSTM-based Seq2Seq model [12] in language modeling to learn video representations (fully connected LSTM: FC-LSTM). Shi *et al.* [13] extended FC-LSTM

This work was supported in part by the National Natural Science Foundation of China (62025101, 61961130392), PKU-Alibaba Fund, and High-performance Computing Platform of Peking University, which are gratefully acknowledged.

978-1-6654-3864-3/21/$31.00 ©2021 IEEE

by applying convolutional operations to LSTMs to learn high-level spatial representations, denoted as Conv-LSTM, which has obtained obvious performance improvement for video prediction. However, both of the LSTM-based video prediction works only focus on modeling temporal variations (such as the object moving trajectories) and ignored the spatial appearance of video frames (such as the perceptual quality for each reconstructed video frame). In [14], spatial deformations and temporal dynamics have been proved equally significant for video prediction, and a new LSTM structure named Spatiotemporal LSTM (PredRNN) was proposed to learn more robust video representations. Wang *et al.* [15] improved PredRNN by solving the gradient propagation difficulties in deep predictive models (PredRNN++). Although PredRNN++ has achieved more promising results with good perceptual quality for each frame in spatial domain, the performance in predicting videos with longer temporal period is still not satisfactory. To predict longer video sequences, the Eidetic 3D LSTM (E3D-LSTM) approach has been proposed in [16], which jointly leveraged the short-term as well as the long-term frame dependencies in videos by applying 3D convolutions to capture better short-term frame dependency and designing a new RECALL gate into PredRNN to capture the long-term temporal dependency. However, as the period of the video sequences to be predicted increasing, the visual quality dramatically degrade and the computation load is high. To reduce the computation load, Yu *et al.* [17] built a Conditionally Reversible Network (CrevNet) and have achieved satisfactory results in next-frame prediction task, however, the quality degradation problem in multi-frame prediction task is still needed to be solved. To generate satisfactory results in multi-frame prediction task, Jin *et al.* [18] utilized multi-frequency information of videos to predict video frames with fine details. However, the computation load is still prohibitively high.

To solve the quality degradation problem in multi-prediction task with an acceptable computation load, we propose an Attention based SpatioTemporal Model (ASTM) for Video prediction by leveraging 3D Convolutional Neural Networks and Long Short-Term Memory (LSTM). In particular, we design an *Efficient Inter-Frame Attention Gate* (EIFAG) to aggregate the multi-term inter-frame dependencies efficiently and utilize 3D convolution operations (Conv3D) to capture more accurate short-term inter-frame dependency. In this way, more reliable motion information can be captured with an acceptable computation load. In addition, we utilize the multi-layer PredRNN structure to extract efficient features in spatial domain to improve the perceptual quality of each predicted video frame. To further evaluate the proposed method on more real scenes, we collect a multi-level spatiotemporal (MLST) dataset from YouTube. Experimental results show that the proposed model can achieve state-of-the-art performance on widely used datasets and the MLST dataset.

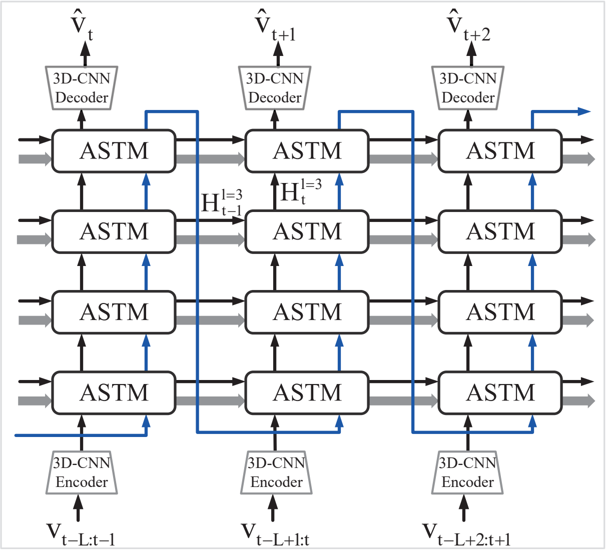

Fig. 1. The SpatioTemporal Recurrent Neural Networks with ASTM. The blue flows denote the appearance information in spatial domain. The gray flows denote the multi-term inter-frame dependencies in temporal domain. The black flows denote the hidden state information.

2. PROPOSED MODEL

In this section, we detailedly introduce the proposed model (ASTM) in capturing both multi-term and short-term inter-frame dependencies in temporal domain as well as learning efficient appearance features in spatial domain for video prediction. The architecture of the proposed model is illustrated in Fig. 1, where a muti-layer Spatiotemporal Recurrent Neural Network is constructed on the basis of ASTM. The blue flows denote memories M in spatial domain which learn efficient appearance representations video frames. The gray flows denote the multi-term inter-frame dependencies C in temporal domain. For each time step, a video clip consisting of a few number of frames will be fed into a single 3D convolutional encoder to extract deep features. Then the deep features will be fed into the corresponding ASTM. Finally, a single 3D convolutional layer is used as the decoder to map the output hidden state to the predicted frame.

2.1. The proposed Attention-based Spatiotemporal Model: ASTM

The detailed structure of the proposed ASTM is illustrated in Fig. 2. There are 4 inputs for the ASTM at time step t $(0 \leq t < T)$ in layer k $(1 \leq k \leq N)$: X_t, the encoded features or hidden states from the previous ASTM layer, i.e., H_t^{k-1}; H_{t-1}^k, the hidden state from the previous time step; $C_{t-\tau:t-1}^k$, the multi-term inter-frame dependencies from previous τ time steps; M_t^{k-1}, the spatial appearance features from the previous ASTM layer. In our method, we jointly leverage temporal

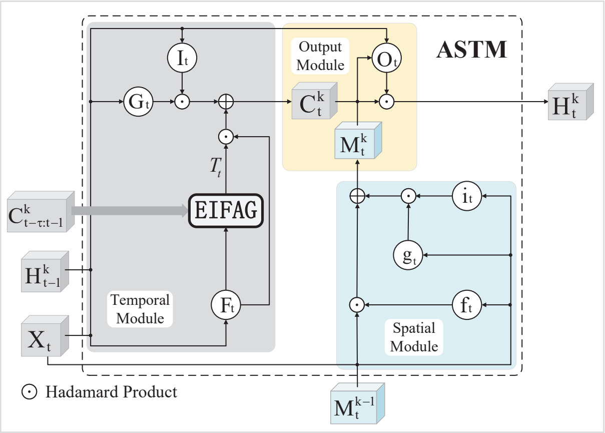

Fig. 2. The overall structure of ASTM.

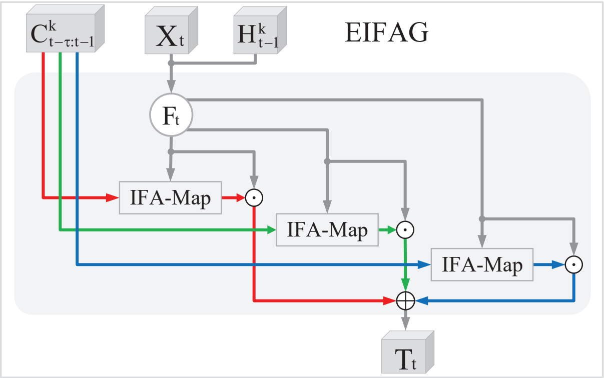

Fig. 3. The structure of the proposed Efficient Inter-Frame Attention Gate (EIFAG).

dependency among video frames and spatial appearance of each frame to learn promising video representations for video prediction. Herein, we jointly utilize both short-term and multi-term inter-frame dependencies to represent the video temporal dependency. In particular, for short-term frame dependency, 3D convolutional operations are utilized to model the state-to-state transitions in ASTMs to improve the local perceptions of the memory cells. These 3D convolutional operations can be formulated as follows,

$$\begin{aligned} X_t &= \begin{cases} Encoder(v_{t-L+1:t}), & k=1 \\ H_t^{k-1}, & otherwise \end{cases} \\ I_t &= \sigma(W_{xi} * X_t + W_{hi} * H_{t-1}^k + b_i), \\ G_t &= \tanh(W_{xg} * X_t + W_{hg} * H_{t-1}^k + b_g), \\ F_t &= \sigma(W_{xf} * X_t + W_{hf} * H_{t-1}^k + b_f), \end{aligned} \tag{1}$$

where $v_{t-L+1:t}$ denotes the video clip input at time step t and L denotes the temporal length of the video clip input. $Encoder(\cdot)$ denotes the 3D convolutional encoder. $\sigma(\cdot)$ is the sigmoid function and $*$ represents the 3D convolutional operation. I_t is the temporal input gate, G_t is the temporal input modulation gate and F_t is the temporal forget gate. W denotes parameters of the integrated 3D convolutional operations and b is the bias variable. To efficiently aggregate multi-term inter-frame dependencies in temporal domain, we design an *Efficient Inter-Frame Attention Gate* (EIFAG) to keep more temporal information from previous temporal memories, which will be introduced in section 2.2 in detail. By jointly utilizing both short-term and multi-term inter-frame dependencies, the proposed model can capture more reliable motion information for video prediction. The aggregated inter-frame dependency C_t^k based on the proposed method can be computed as follows,

$$C_t^k = I_t \odot G_t + F_t \odot T_t, \tag{2}$$

where $\odot$ is the Hadamard product. The first term $I_t \odot G_t$ denotes the encoded temporal information of current video clip input. The second term $F_t \odot T_t$ denotes the preserved inter-frame dependency from previous time τ time steps. In particular, T_t denotes the aggregated multi-term inter-frame dependency in temporal domain and is defined as follows,

$$T_t = EIFAG(C^k_{t-\tau:t-1}, F_t). \tag{3}$$

T_t consists of two terms, where the first term denotes the multi-term inter-frame dependencies and the second term denotes the temporal forget gate which can represent the current input. By aggregate both terms with the proposed EIFAG, the most relative temporal information to current input can be efficiently extracted from previous τ time steps. The overall structure for temporal information acquisition is shown as the gray module in Fig. 2. To learn efficient features in representing spatial appearance of each frame, we utilize the multi-layer PredRNN structure, and the spatial state-to-state transitions are described in detail as follows:

$$\begin{aligned} i_t &= \sigma(w_{xi} * X_t + w_{mi} * M_t^{k-1} + b_i'), \\ g_t &= \tanh(w_{xg} * X_t + w_{mg} * M_t^{k-1} + b_g'), \\ f_t &= \sigma(w_{xf} * X_t + w_{mf} * M_t^{k-1} + b_f'), \\ M_t^k &= M_t^{k-1} \odot f_t + i_t \odot g_t. \end{aligned} \tag{4}$$

The proposed spatial information M_t^k is computed with two terms as traditional LSTMs, and the overall structure for spatial information acquisition is illustrated as the blue module in Fig. 2. The first term $M_t^{k-1} \odot f_t$ denotes the preserved spatial appearance information, where f_t denotes the spatial forget gate. The second term $i_t \odot g_t$ denotes the encoded spatial appearance information, where i_t denotes the spatial input gate and g_t denotes the spatial input modulation gate. w represents parameters of the integrated 3D convolutional operations and b' is the bias variable. By jointly utilizing the proposed temporal dependency C_t^k and spatial features M_t^k, we can get the final output and hidden state for current time step:

$$\begin{aligned} O_t^k &= \sigma(W_{xo} * X_t + W_{ho} * H_{t-1}^k \\ &\quad + W_{mo} * M_t^k + W_{co} * C_t^k + b_o), \\ H_t^k &= O_t^k \odot \tanh(W_{1\times1\times1} * [C_t^k, M_t^k]), \end{aligned} \quad (5)$$

where O_t^k denotes the spatiotemporal output gate and H_t^k denotes the final hidden state, which are shown as the orange module in Fig. 1. By applying a single 3D convolutional decoder at the end of the model, the predicted video clips at time step t can be retained:

$$\hat{v}_{t+1} = Decoder(H_t^N), \quad (6)$$

where $\hat{v}_{t+1}$ denotes the predicted frame at time step t, and N denotes the total number of ASTM layers.

2.2. Efficient Inter-Frame Attention Gate (EIFAG)

To aggregate the multi-term inter-frame dependencies in temporal domain with an acceptable computation load, we design an *Efficient Inter-frame Attention Gate* (EIFAG), which is partially motivated by the attention based machine translation scheme for long sequences in [19]. The detailed structure of EIFAG is illustrated in Fig. 3.

In traditional LSTMs, the forget gate F_t is actually a feature map with values ranging from 0 to 1. The feature map reflects the percentage of the preserved information from current inputs. While applying Hadamard product to F_t and the previous temporal memory C_{t-1}^k, the LSTMs can selectively preserve the most useful information from the previous temporal memory C_{t-1}^k. However, when the sequences are too long, the LSTMs need to remember more information from more previous temporal memories, i.e., $C_{t-\tau:t-1}^k$ instead of merely C_{t-1}^k, to predict a more reliable future. Thus, we define an *Inter-frame Attention Map* (IFA-Map) operation for EIFAG, which can reflect the correlations between two memory states from different time steps. By applying IFA-Map operation to F_t and the multiple previous temporal memories $C_{t-\tau:t-1}^k$ respectively, we can get a new *multi-forget gate* with the same shape as $C_{t-\tau:t-1}^k$. By jointly utilizing the *multi-forget gate* and the previous temporal memory C_{t-1}^k, we can obtain the aggregated multi-term inter-frame dependency T_t, which can reflect the most relevant temporal information to current input. The EIFAG calculation can be formulated as follows,

$$T_t = \sum_{i=1}^{\tau} (\textit{IFA-Map}(C_{t-i}^k, F_t) \odot C_{t-i}^k), \quad (7)$$

where *IFA-Map*$(\cdot)$ is a combination of Hadamard product and softmax function, as shown in Eq.(8).

$$\textit{IFA-Map}(C_{t-i}^k, F_t) = \frac{e^{C_{t-i}^k \odot F_t}}{\sum_{i=1}^{\tau}(e^{C_{t-i}^k \odot F_t})}. \quad (8)$$

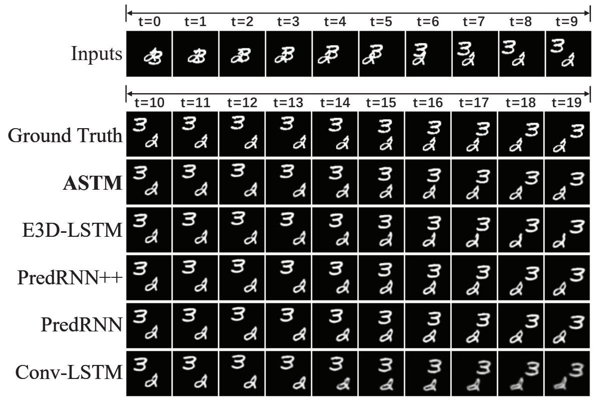

Fig. 4. Comparisons of the generated examles on the Moving MNIST test set (10 frames → 10 frames).

3. EXPERIMENTS

In this section, we evaluate the proposed model in three datasets, the Moving MNIST dataset [11], the KTH action dataset [20] and the proposed MLST dataset. We set $\tau = 5$ in our experiments and optimize our model with MSE loss using ADAM optimizer. All experiments are conducted using Pytorch and the source code will be released to the public. We stack 4 ASTMs in our model and the integrated 3D convolutional operators are set with a kernel size $3 \times 5 \times 5$ (time, height, width), and the stride is set to 1 for each dimension. The number of hidden state channels for each ASTM is 64.

3.1. Moving MNIST dataset

We first use the Moving MNIST dataset generated in [11] to evaluate the baseline performance of our model. Each Moving MNIST sequence consists of 20 frames with 64×64 pixels. We use 10,000 sequences for training, and 3,000 for testing, in particular we use 10 frames to predict the next 10 frames. In Fig. 4, we illustrate the visual results of predicted videos from the proposed method and other state-of-the-art methods. We can see that our method obviously outperforms the state-of-the-art methods in visual quality with more clear digits. In particular, the digit boundaries of the last three predicted frames become blurry in other methods while our model can still predict visually pleasant results. Table 1 shows quantitative results of the proposed method and other state-of-the-art methods, where we use MSE (Mean Square Error) and SSIM (Structural Similarity) [21] score to measure the visual quality of the predicted results. As shown in Table 1, ASTM significantly outperforms others in SSIM and MSE score and the GPU memory usage is lower than most of the methods, which indicates the proposed methods can predict multiple high-quality frames with an acceptable computation load.

Table 1. Quantitative results of different methods on the Moving MNIST test set. Lower MSE score, and higher SSIM scores indicate better prediction results. The results of the compared methods are reported in [17].

Moving MNIST			
Method	SSIM	MSE (10^{-4})	Memory (1 sample)
Conv-LSTM [13]	0.707	103.3	-
FRNN [22]	0.819	68.4	717MB
VPN [23]	0.870	70.0	5206MB
PredRNN [14]	0.869	56.8	-
PredRNN++ [15]	0.898	46.5	-
E3D-LSTM [16]	0.910	41.3	2695MB
CrevNet + ConvLSTM [17]	0.928	38.5	**130MB**
ASTM	**0.944**	**37.2**	225MB

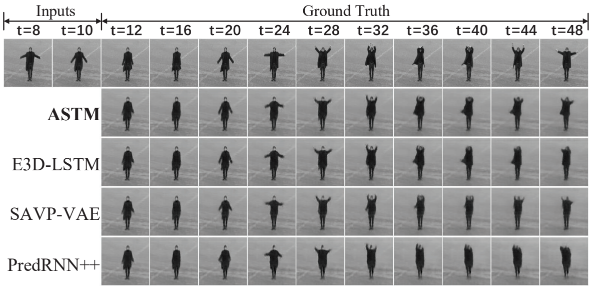

Fig. 5. Comparisons of the generated examples on the KTH action test set (10 frames → 40 frames).

3.2. KTH action dataset and MLST dataset

The KTH action dataset contains 25 individuals performing 6 types of actions, including walking, jogging, running, boxing, hand waving and hand clapping. On average, each video clip lasts 4 seconds. We follow the experiment settings in [14] and each frame is resized to 128×128 pixels. We train the proposed model to predict the next 10 frames with 10 frames as the input. While testing, the period of the predicted videos is extended to 20 and 40. To evaluate the proposed model on more complex real video data, we also collect a multi-level spatiotemporal (MLST) dataset from Youtube with a resolution 240×320. The MLST dataset contains diverse kinds of videos in order to approximate the real environment, which can be divided into 9 categories based on the spatiotemporal complexity in videos and each frame is resized to 256×256 pixels. We retrained the state-of-the-art models, including Conv-LSTM [13], PredRNN [14], PredRNN++ [15] and E3D-LSTM [16], on the proposed MLST dataset using the official code. All models are trained to predict the next 20 frames with 10 successive frames as the input.

Fig. 5 and Fig. 6 show the qualitative results of ASTM and other state-of-the-art methods, where the proposed method significantly outperforms others and generates the most realistic results on both datasets. Table 2 shows quantitative results of the proposed model and the compared methods, where ASTM achieves the best quantitative results compared to state-of-the-art methods.

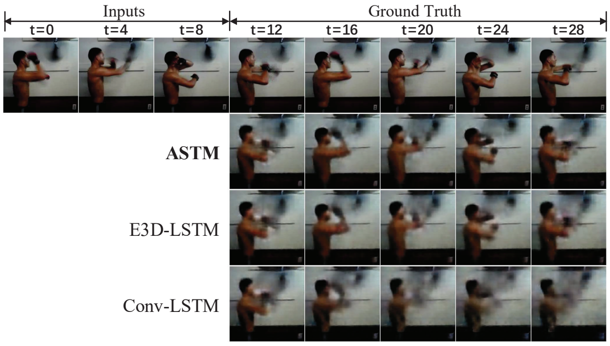

Fig. 6. Comparisons of the generated examles on the MLST test set (10 frames → 20 frames).

Table 2. Quantitative results of different methods on the KTH action and MLST test set. Higher PSNR and higher SSIM scores indicate better prediction results. The results of the compared methods are reported in [24].

Method	KTH				MLST
	10 → 20		10 → 40		10 → 20
	SSIM	PSNR	SSIM	PSNR	PSNR
ConvLSTM [13]	0.712	23.58	0.639	22.85	21.35
FRNN [22]	0.771	26.12	0.678	23.77	-
PredRNN [14]	0.839	27.55	0.703	24.16	22.90
PredRNN++ [15]	0.865	28.47	0.741	25.21	23.16
SAVP-VAE [6]	0.852	27.77	0.811	26.18	-
E3D-LSTM [16]	0.879	29.31	0.810	27.24	23.37
Jin *et al.* [18]	0.893	29.85	0.851	27.56	-
ASTM	**0.908**	**30.02**	**0.865**	**28.12**	**24.73**

Table 3. Ablation Study for ASTM.

Method	Moving MNIST		KTH action		MLST
	10 → 10		10 → 40		10 → 20
	MSE(10^{-4})	SSIM	PSNR	SSIM	PSNR
Basic Model	45.4	0.905	25.58	0.789	23.21
ASTM	**36.2**	**0.936**	**28.12**	**0.865**	**24.73**

3.3. Ablation Study

To evaluate the efficiency of the proposed Efficient Inter-Frame Attention Gate (EIFAG) in ASTM, we conduct a series of ablation studies. We evaluate the performance compared with the basic model (ASTM w/o EIFAG) and summarize the quantitative results in Table 3. As shown in Table 3, ASTM obviously outperforms the basic model, indicating the proposed EIFAG can help ASTM to learn a more promising spatiotemporal representation for videos.

4. CONCLUSION

In this paper, we proposed an Attention based SpatioTemporal Model (ASTM) for video prediction. In our method, we jointly utilized temporal and spatial information to learn efficient spatiotemporal representations for videos. We first designed an *Efficient Inter-frame Attention-based Gate* (EIFAG) to aggregate the multi-term inter-frame dependencies in temporal domain with an acceptable computation load, and then utilized 3D convolutional operations to capture better short-term inter-frame dependency. By further leveraging multi-layer PredRNN structure, efficient spatial appearance information of each video frame can be learned. Experimental results show that the proposed methods outperforms diverse state-of-the-arts in all datasets.

5. REFERENCES

[1] Siwei Ma, Xinfeng Zhang, Chuanmin Jia, Zhenghui Zhao, Shiqi Wang, and Shanshe Wang, "Image and video compression with neural networks: A review," *TCSVT*, 2019.

[2] Ivan Zupancic, Saverio G Blasi, Eduardo Peixoto, and Ebroul Izquierdo, "Inter-prediction optimizations for video coding using adaptive coding unit visiting order," *TMM*, vol. 18, no. 9, pp. 1677–1690, 2016.

[3] Chelsea Finn, Ian Goodfellow, and Sergey Levine, "Unsupervised learning for physical interaction through video prediction," in *NeurIPS*, 2016, pp. 64–72.

[4] Michael Mathieu, Camille Couprie, and Yann LeCun, "Deep multi-scale video prediction beyond mean square error," in *ICLR*, 2016.

[5] Mohammad Babaeizadeh, Chelsea Finn, Dumitru Erhan, Roy H Campbell, and Sergey Levine, "Stochastic variational video prediction," in *ICLR*, 2018.

[6] Alex X Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine, "Stochastic adversarial video prediction," *arXiv preprint arXiv:1804.01523*, 2018.

[7] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh, "Action-conditional video prediction using deep networks in atari games," in *NeurIPS*, 2015, pp. 2863–2871.

[8] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[9] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP*, 2014.

[10] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP*, 2013, pp. 6645–6649.

[11] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov, "Unsupervised learning of video representations using lstms," in *ICML*, 2015, pp. 843–852.

[12] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," in *NeurIPS*, 2014, pp. 3104–3112.

[13] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo, "Convolutional lstm network: a machine learning approach for precipitation nowcasting," in *NeurIPS*, 2015, pp. 802–810.

[14] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S Yu, "Predrnn: recurrent neural networks for predictive learning using spatiotemporal lstms," in *NeurIPS*, 2017, pp. 879–888.

[15] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and S Yu Philip, "Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning," in *ICML*, 2018, pp. 5123–5132.

[16] Yunbo Wang, Lu Jiang, Ming Hsuan Yang, Li Jia Li, Mingsheng Long, and Li Fei-Fei, "Eidetic 3d lstm: A model for video prediction and beyond," in *ICLR*, 2019.

[17] Wei Yu, Yichao Lu, Steve Easterbrook, and Sanja Fidler, "Efficient and information-preserving future frame prediction and beyond," in *ICLR*, 2019.

[18] Beibei Jin, Yu Hu, Qiankun Tang, Jingyu Niu, Zhiping Shi, Yinhe Han, and Xiaowei Li, "Exploring spatial-temporal multi-frequency analysis for high-fidelity and temporal-consistency video prediction," in *CVPR*, 2020, pp. 4554–4563.

[19] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[20] Christian Schuldt, Ivan Laptev, and Barbara Caputo, "Recognizing human actions: a local svm approach," in *ICPR*, 2004, vol. 3, pp. 32–36.

[21] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[22] Marc Oliu, Javier Selva, and Sergio Escalera, "Folded recurrent neural networks for future video prediction," in *ECCV*, 2018, pp. 716–731.

[23] Nal Kalchbrenner, Aäron Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu, "Video pixel networks," in *ICML*, 2017, pp. 1771–1779.

[24] Sergiu Oprea, Pablo Martinez-Gonzalez, Alberto Garcia-Garcia, John Alejandro Castro-Vargas, Sergio Orts-Escolano, Jose Garcia-Rodriguez, and Antonis Argyros, "A review on deep learning techniques for video prediction," *TPAMI*, 2020.