Neurocomputing 331 (2019) 235–241

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Deep memory and prediction neural network for video prediction

Zhipeng Liu[a,c], Xiujuan Chai[a,b,*], Xilin Chen[a,c]

[a] Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing 100190, China
[b] Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing 100081, China
[c] University of Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

Article history:
Received 29 December 2017
Revised 31 October 2018
Accepted 19 November 2018
Available online 22 November 2018

Communicated by Wei Wu

Keywords:
Video prediction
Predictive error
Memory mechanism
Neural network

ABSTRACT

Inspired by the concept of memory mechanism and predictive coding from the cognitive neuroscience, this paper presents a deep memory and prediction neural network (DMPNet) for video prediction. Correspondingly, memory and error propagation units are designed in DMPNet to capture the previous spatial-temporal information and compute current predictive error which is forwarded to the prediction unit for correcting the subsequent video prediction. Subsequently, prediction unit takes the information stored in memory unit and predictive error of previous frame as input to predict the next frame. We evaluate our method on two public real-world datasets and demonstrate that the proposed DMPNet outperforms some state-of-the-art methods quantitatively and qualitatively.

1. Introduction

In recent years, visual future prediction has gained great attention because of its potential applications, such as autonomous cars, robot, video understanding. In addition, the learned internal feature representations from video prediction can be used for many supervised tasks, such as action recognition [1,2], semantic segmentation [3] and optical flow estimation [4] and so on. It has led to a variety of tasks due to different goals of future prediction, such as, human activity [5,6] and event [7]. However, these tasks are high-level semantic predictions and require fully-labeled training data. To alleviate this issue, some researchers in deep learning community have targeted pixel-level video prediction with unsupervised learning. The task is to predict the next short frames when given a sequence of previous frames. Actually, video prediction is an extremely challenging problem, as it is difficult to learn the complicated spatial-temporal correlation in real-world videos.

Many methods have been developed to deal with the video prediction. Most of them are based on the encoder-decoder framework. These methods can be roughly divided into two categories. One models the motion pattern implicitly and decodes future frames directly from scratch. The other pays attention to model the motion pattern explicitly. Ranzato et al. [8] propose a baseline for video prediction at the beginning. They use a convolutional recurrent neural network to encode the quantized patches in a frame and decode the central patches of the next predicted frame. Srivastava et al. [1] apply a sequence-to-sequence learning framework for video prediction and show that long-short time memory (LSTM) networks are capable of predicting the motion of bouncing handwritten digits. This work has been extended by Shi et al. [9] who propose to use convolutional LSTM to replace the fully connected LSTM in the network. The use of convolutional LSTM reduces the number of parameters and exploits the local spatial correlation of image. Then some methods are proposed in order to employ motion priors, such as the direction of movement of object in videos. Oh et al. [10] present the encoding-transformation-decoding framework to address the problem of predicting future frames in Atari games conditioned on both previous frames and actions. Prior action information are integrated into their transformation component directly. Moreover, Villegas et al. [11] first employ the prior motion of human skeleton points and design a hierarchical network for pixel-level prediction. Their framework independently predicts skeleton points and generates the future frames. Nevertheless, their method can only be used in the constrained scenarios which must contain the human. Except for encoding the time and space, Kalchbrenner et al. [12] formulate the time, space and color structure as a four-dimensional dependency chain and achieve the best performance on the synthesis data, Moving Mnist. However, the network has a huge number of

[*] Corresponding author at: Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing 100190, China.

E-mail addresses: zhipeng.liu@vipl.ict.ac.cn (Z. Liu), xiujuan.chai@vipl.ict.ac.cn (X. Chai), xilin.chen@vipl.ict.ac.cn (X. Chen).

https://doi.org/10.1016/j.neucom.2018.11.065
0925-2312/© 2018 Elsevier B.V. All rights reserved.

parameters and runs slowly because it uses about 80 layers and predicts pixel by pixel. As for the second category, De Brabandere et al. [13] design dynamic filters network (DFN) for video prediction. The filters predict local pixel transformation in similar to optical flow. The model described in Patraucean et al. [3] encodes motion pattern explicitly by predicting a dense flow map. Finn et al. [14] use convolutional LSTM and motion models to predict a distribution over pixel motion from previous frames in action-conditioned video prediction.

Apart from encoder-decoder framework, Lotter et al. [15] propose a predictive neural network (PredNet) inspired by the concept of predictive coding from the cognitive neuroscience. PredNet forwards the predictive error to improve the subsequent video prediction. It employs both bottom-up and top-down connections and achieves great performance in real-world datasets. However, there is a problem in PredNet. Their network only uses predictive error to predict frames and does not have a memory unit to remember previous frames, which is different from the process of human cognition. Some researches in cognitive neuroscience show that memory also plays a great role in the process of cognition [16–19].

In the field of cognitive neuroscience, memory mechanism and predictive coding are important concepts [20]. Hawkins et al. [21] propose a sequence memory mechanism that the neocortex may use to store sequences of patterns and recall sequences for making predictions and recognizing time-based patterns. As for predictive coding, it assumes that the brain is continually making predictions of incoming sensory stimuli [22]. Top-down (or lateral) connections convey these predictions which are compared against actual observations to generate an error signal (named as predictive error). The error signal is then propagated back up the hierarchy to correct the subsequent predictions. They depict two crucial aspects of human cognition, which are how to employ the history information and improve the ability of prediction based on predictive error. Inspired by this, a deep memory and prediction neural network is proposed for video prediction. In our neural network, memory and error propagation units are designed to imitate memory and predictive error propagation process of human cognition respectively. Memory unit captures the previous spatial-temporal information. Then error propagation unit computes predictive error between the current estimate of the targeted frame and its top-down prediction from a higher level. Then, the predictive error is forwarded to the prediction unit. Finally, Prediction unit takes the previous information and predictive error as inputs to generate the next frame. The whole model is similar to the process of human cognition [23–25], which not only remembers previous video frames but also employs the predictive error to correct subsequent predictions. In comparison, PredNet only formulates the predictive error and DFN is a simple encoder-decoder method, which is less interpretable. The advantage of our proposed DMPNet is to consider the influence of memory and predictive error together on video prediction and it is more explanatory. In addition, DMPNet achieves much better performance than PredNet and DFN. The contribution of our work mainly lies in three-folds. Firstly, we propose DMPNet and verify that the memory mechanism and predictive error are beneficial for video prediction task. Secondly, we design different type of memory units for different situations. Especially, a simple but effective convolutional memory unit is developed for small datasets. Thirdly, the proposed DMPNet is evaluated on two challenging real-world video datasets and outperforms the state-of-the-art methods.

The remainder of the paper is organized as follows. Section 2 introduces an overview of the novel DMPNet. Section 3 gives the detail of the proposed network. Experimental results on real-world video prediction are illustrated in Section 4. Finally, we conclude the paper.

2. Overview of DMPNet

In this section, we formulate the problem of video prediction and explain the role of each unit in the proposed architecture. In general, there are one-step and multi-step video prediction tasks. Let $x_k \in R^{c \times w \times h}$ denote the kth frame in an input video x, where c, w and h denote number of channels, width and height, respectively. In the process of one-step prediction, our network observes a history of previous consecutive frames up to frame k and generates the next frame $\hat{x}_{k+1}$ as follows:

- Memory unit takes previous consecutive frames to frame k as input and captures the spatio-temporal information by using convolutional recurrent layers (ConvRNN). The output is forwarded to the prediction unit.
- Prediction unit is a crucial component, which takes previous spatial-temporal information and predictive error (predictive error is initialized with zeros at the first step) as input and produces the next frame in the lowest layer.
- Error propagation unit computes the predictive error between the current estimate of targeted frame and its top-down prediction from a higher level. Then the predictive error is forwarded to prediction unit and can be used to correct the subsequent video predictions.

As for the multi-step prediction, $\hat{x}_{k+1:k+m}$ can be achieved by taking current predictions as actual next step target frame and recursively iterating m times in both training and testing process. For example, predicted $\hat{x}_{k+1}$ is considered as the true frame x_{k+1}, and the target is to generate the next frame $\hat{x}_{k+2}$.

3. DMPNet for video prediction

This section describes the detailed DMPNet, including the principle of DMPNet, memory unit and loss function.

3.1. The principle of DMPNet

Fig. 1 illustrates an example of DMPNet with two layers. Actually, DMPNet can have different stacked layers according to different situations. It is made up of three kinds of units, including memory (M), prediction (P) and error propagation (E) units. Fig. 1(b) illustrates the module detailed operations for video prediction corresponding to Fig. 1a. The memory unit consists of stacked convolutional recurrent layers and takes previous frames as input. The output of M is forwarded to both its next layer and the corresponding prediction unit. The prediction unit is made up of stacked convolutional LSTM (CLSTM) [9] layers. It takes the output of memory unit, predictive error (the first step predictive error is initialized with zero) and its top-down output as input to generate a prediction. Then the last error propagation unit computes the positive and negative errors, which are passed through a convolutional layer and are the input of the next layer. Meanwhile, P also receives a copy of the error signal. The whole model not only captures the previous spatial-temporal information but also adopts the predictive error to correct the subsequent video prediction.

$$M_l^{(k)} = \begin{cases} x_k & if\ l = 1 \\ ConvRNN(M_{l-1}^{(k)}, M_l^{(k-1)}) & l > 1 \end{cases} \tag{1}$$

$$P_l^{(k+1)} = \begin{cases} CLSTM(M_l^{(k)}, E_l^{(k)}, P_l^{(k)}) & if\ l = L \\ CLSTM(M_l^{(k)}, E_l^{(k)}, P_l^{(k)}, UpSample(P_{l+1}^{(k+1)})) & l < L \end{cases} \tag{2}$$

$$R_l^{(k+1)} = \begin{cases} x_{k+1} & if\ l = 1 \\ MaxPooling(ReLU(Conv(E_{l-1}^{(k)}))) & l > 1 \end{cases} \tag{3}$$

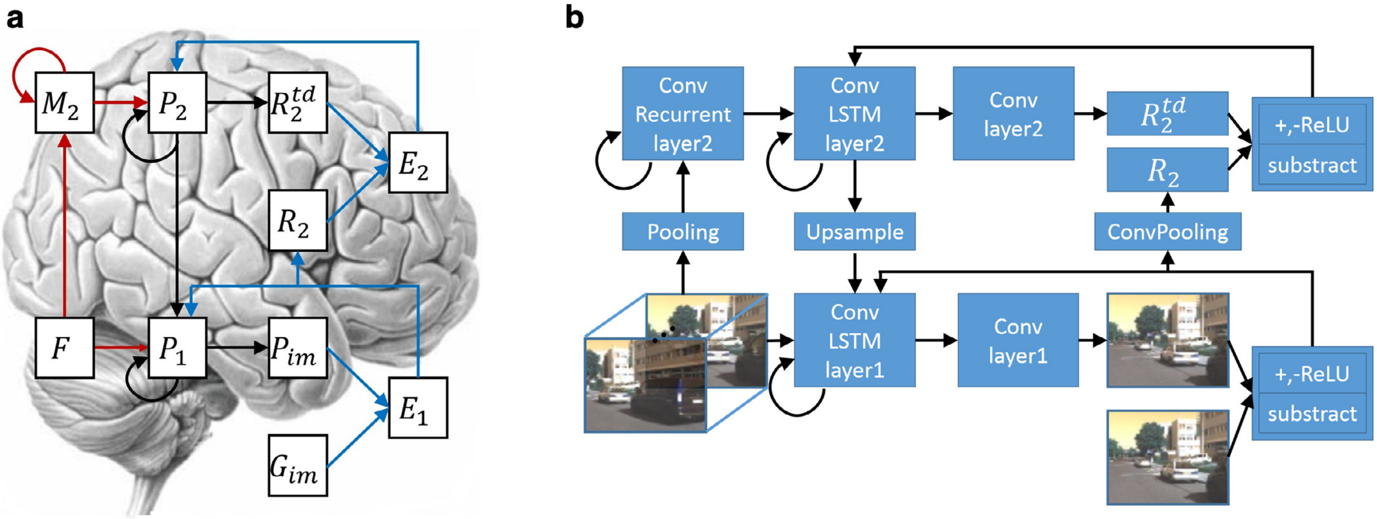


Fig. 1. Deep Memory and Prediction Neural Network (DMPNet) for video prediction. (a) Illustration of information flow with two layers. DMPNet consists of memory unit (*M*, previous frames are considered as a part of memory unit directly in the lowest layer), prediction unit (*P*) and error propagation unit (*E*). *R* denotes the responses which maintain the current estimation of the targeted frame. *F, P, R^{td}* and *G* denote previous frames, one prediction frame, a top-down prediction from a higher level and ground truth frame, respectively. Red connections denote the information flow of the memory unit, which captures the previous spatial-temporal information. Black connections denote the information flow of the prediction unit, which generates the responses (the lowest layer response represents the predicted image). Blue connections represent the information flow of the error propagation unit, whose output is forwarded to the prediction unit and can be used to correct the subsequent video prediction. (b) Detailed module operations for video prediction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$R_l^{td(k+1)} = \begin{cases} \min(p_{\max}, ReLU(Conv(P_l^{(k+1)}))) & if\ l = 1 \\ ReLU(Conv(P_l^{(k+1)})) & l > 1 \end{cases} \tag{4}$$

$$E_l^{(k+1)} = \left[ReLU(R_l^{(k+1)} - R_l^{td(k+1)}), ReLU(R_l^{td(k+1)} - R_l^{(k+1)})\right] \tag{5}$$

$$\hat{x}_{k+1} = R_1^{td(k+1)} \tag{6}$$

The full set of update principle is listed in Eqs. (1)–(6), where L denotes the number of layers, $ReLU(\cdot)$ is rectified linear unit activation, $[\cdot, \ \cdot]$ denotes channel-wise concatenation. DMPNet generates the next frame x_{k+1}, given the previous sequence of frames $x_{1:\,k}$. The memory unit takes $x_{1:\,k}$ as input one by one and captures the previous spatial-temporal information, which consists of stacked ConvRNN layers as shown in Eq. (1). The output $M_l^{(k)}$ is forwarded to its next layer and corresponding prediction unit with lateral connection. For the prediction unit, we empirically use stacked CLSTM layers. The prediction unit produces $P_l^{(k+1)}$ according to $M_l^{(k)}$, $E_l^{(k)}$, $P_l^{(k)}$, as well as $P_{l+1}^{(k+1)}$ which is spatial upsampled (bilinear) as shown in Eq. (2). After predicting the next frame, the targeted response $R_1^{(k+1)}$ for the first layer is set to the ground truth frame and the targeted responses for higher layers are computed by a convolutional layer over the predictive error $E_{l-1}^{(k)}$ from the layer below, followed by $ReLU(\cdot)$ and max-pooling as shown in Eq. (3). The max-pooling can amplify the receptive-field. The targeted response is designed to compute the higher level features as the predictive error propagates up the network [26]. The top-down prediction $R_l^{td(k+1)}$ is computed by a convolution of the $P_l^{(k+1)}$ followed by a $ReLU(\cdot)$. For the lowest layer, $P_l^{(k+1)}$ is also passed through a max-clipped non-linearity as shown in Eq. (4). Then, the predictive error is computed as shown in Eq. (5), which is split into position and negative errors. The separate error populations are analogous to the existence of on-center, off-surround and off-center, on-surround neurons early in the visual system [15]. $E_l^{(k+1)}$ is forwarded to *P* and can be used to correct the subsequent video prediction. Finally, We obtain the predicted frame that is the lowest top-down prediction frame $R_1^{td(k+1)}$. The video prediction procedure of DMPNet is described succinctly in Algorithm 1. DMPNet imitates the retention of long-term memories and predictive error in the process of human cognition by using memory and error propagation unit respectively. The memory unit can directly capture the previous spatial-temporal information and error propagation unit is used to compute the predictive error.

Algorithm 1 DMPNet for Video Prediction.

```
Input: a sequence of frames $x_{1:k}$, the number of layer $L$
Output: the next predicted frame $\hat{x}_{k+1}$
$M_{1:l}^{(0)}, E_{1:l}^{(0)}, P_{1:l}^{(0)} \leftarrow 0$
$R_1^{(1:t)} \leftarrow x_{1:t}$
for t = 1 to k do
    for $l$ = 1 to L do
        if $l$ = 1 then
            $M_l^{(t)} \leftarrow x_t$                                    ▷ Insert.
        else
            $M_l^{(t)} \leftarrow ConvRNN(M_{l-1}^{(t)}, M_l^{(t-1)})$
    for $l$ = L to 1 do
        if $l$ = L then
            $P_l^{t+1} \leftarrow CLSTM(M_l^{(t)}, E_l^{(t)}, P_l^{(t)})$
        else
            $P_l^{t+1} \leftarrow CLSTM(M_l^{(t)}, E_l^{(t)}, P_l^{(t)}, UpSample(P_{l+1}^{(t+1)}))$
    for $l$ = 1 to L do
        if $l$ = 1 then
            $R_l^{td(t+1)} \leftarrow \min(p_{\max}, ReLU(Conv(P_l^{(t+1)})))$
        else
            $R_l^{td(t+1)} \leftarrow ReLU(Conv(P_l^{(t+1)}))$
        $E_l^{(t+1)} \leftarrow [ReLU(R_l^{(t+1)} - R_l^{td(t+1)}), ReLU(R_l^{td(t+1)} - R_l^{(t+1)})]$
        if $l$ < L then
            $R_{l+1}^{(t+1)} \leftarrow MaxPooling(ReLU(Conv(E_{l-1}^{(t)})))$
return $R_1^{td(k+1)}$
```

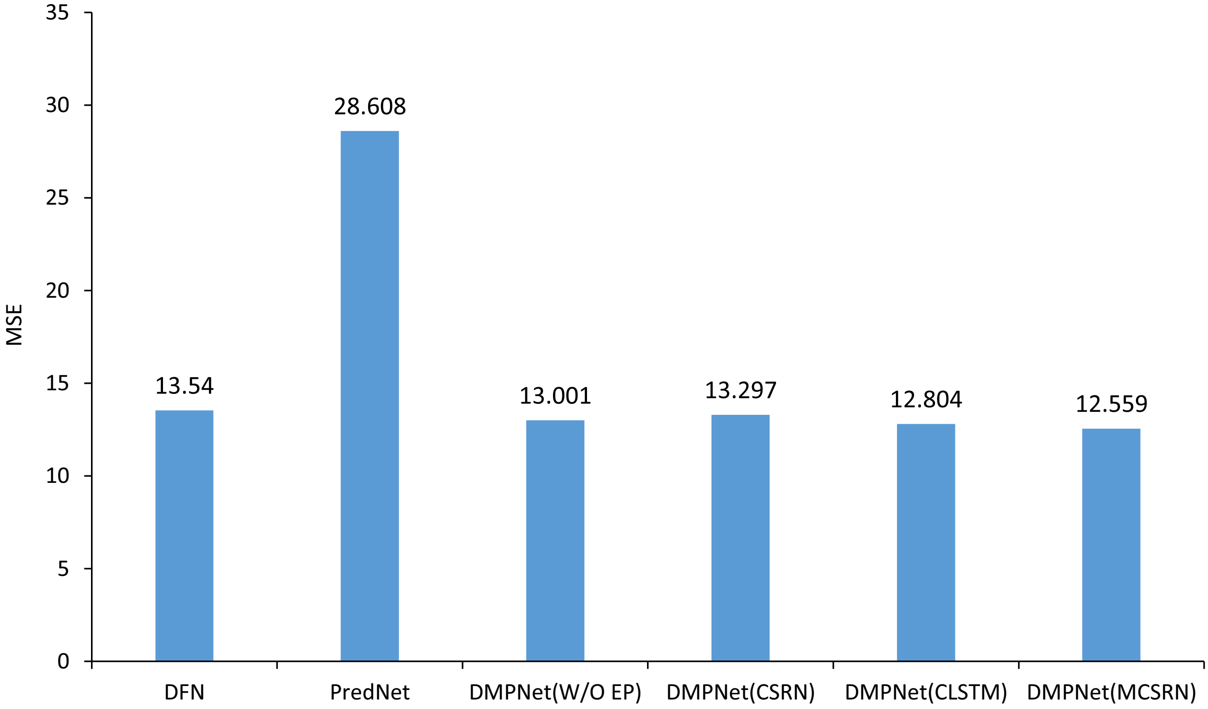


Fig. 2. Quantitative comparison of different methods on highway driving dataset, the lower the better.

They both play the roles in improving the performance of video prediction.

3.2. Memory unit

The prediction and error propagation units have been described in [15]. So we deeply explore the design principle of the memory unit in this section. The target of this unit is to capture the spatio-temporal correlations among the previous sequential frames. As a powerful tool to investigate the contextual correlation, LSTM has gained great success in many fields, such as speech recognition [27], language translation [28], handwriting recognition [29] and so on. Therefore, convolutional LSTM (CLSTM) can be adopted in our memory unit in term of video prediction. The CLSTM is widely used to capture the spatio-temporal correlations [11]. However, when confronted with small datasets, it is difficult to train the huge number of CLSTM's parameters and CSLTM is easily over-fitting. Simple but effective memory unit is needed for small and simple datesets. We adopt convolutional simple recurrent networks (CSRN), which only has one gate and fewer parameters than CLSTM. CSRN is described as follows:

$$h_t = H(W_{zh} * z + W_{hh} * h_{t-1} + b_h) \tag{7}$$

$$y_t = O(W_{oh} * h_t + b_o), \tag{8}$$

where z is the input, h_{t-1} is the previous hidden state, all the vectors b denote the bias terms, all the matrices W are convolutional-kernel tensors, * is the convolution operator, $H(\cdot)$ and $O(\cdot)$ are activation functions in the hidden layer and output layer. In order to reduce the parameters of SRN further, we replace Eq. (8) with Eq. (9) and call the modified CSRN as MCSRN for short.

$$y_t = h_t \tag{9}$$

3.3. Loss function

The task of video prediction is to predict future frames $\hat{x}_{k+1:k+m} (m >= 1)$ given the previous frames $x_{1:\,k}$ by learning the motion pattern. We find that the motion pattern is nearly constant across time in a short video through lots of observation. For example, if a car runs on the highway, the opposite objects will always move towards the car from the view of its car-mounted camera. The prediction model is mainly to learn the motion pattern that people design the loss function. Previous loss functions mainly focus on the difference between predicted $\hat{x}_{k+1:k+m}$ and ground truth $x_{k+1:k+m}$ to learn the latter motion pattern in a video. We call it as frame loss. Because of the consistency of movement, differences between $\hat{x}_{1:k}$ and $x_{1:\,k}$ can also be used to learn the front motion pattern. We call this differences as model loss. As for our model, the predictive error of the lowest layer, which ranges from 1 to k, can be directly used as the model loss. Our final loss function consists of two parts:

$$L = L_{model} + L_{frame}, \tag{10}$$

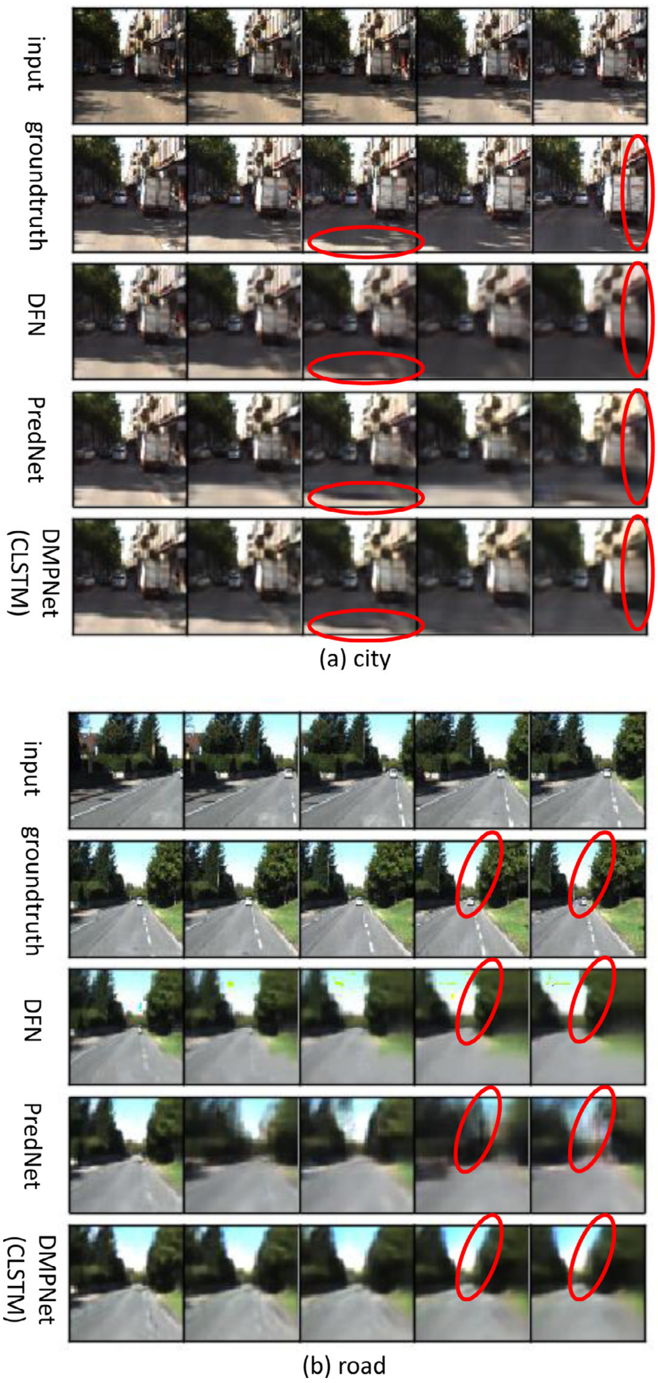
where L_{model} denotes the model loss which is computed by Eq. (5), L_{frame} denotes the frame loss. In order to compare with recent methods fairly, we use L2-based metric to compute L_{frame} in Highway Driving dataset and L1-based metric in KITTI dataset since these methods use different metrics in the two datasets.

4. Experiments

To evaluate the performance of the proposed DMPNet, this section gives the convincing experiments conducted on two public and challenging datasets, which are Highway Driving [13] and KITTI [30]. First, a brief introduction on the experiment settings is presented. Then we give the experimental results and analysis, including the comparison with the state-of-the-art methods, i.e. dynamic filter networks (DFN) [13] and PredNet [15].

4.1. Experiment settings

The DMPNet consists of 4 layers in the following experiment (including the pixel layer) with 3x3 filter sizes for all



(a) city

(b) road

Fig. 3. Qualitative results of multi-step video prediction on KITTI dataset. Five successive images in the first line of (a) and (b) are the input of neural networks and the following four lines are the groundtruth, predicted results of DFN, PredNet and DMPNet(CLSTM), respectively.

convolutions and stack channel sizes per layer of (n, 48, 96, 192), where n denotes the input channel of image. All of the convolutions in the network are zero-padded such that the spatial size is constant within layer across time. Both the stride of maxpooling and the scale factor of bilinear upsampling are 2. All model weights are initialized by using the method proposed in [31] and optimized using the Adam algorithm [32] with default values ($\beta_1 = 0.9$, $\beta_2 = 0.999$). The learning rate is initially set to 0.001, which decreases by a factor of halfway through training until 0.00001. The proposed architecture is implemented using the PyTorch [33].

4.2. Evaluation on highway driving

We first evaluate the proposed DMPNet on a natural video prediction dataset, Highway Driving, which records a car driving on the highway by using a car-mounted camera. It has roughly 20,000 frames and is split into a training set of 16,000 frames and a test set of 4000 frames. This video prediction dataset is challenging because the videos span a wide range of settings and are characterized by rich temporal dynamic information, including both self-motion of the car and the motion of other objects in the scene.

For our proposed DMPNet, there are three variations according to different memory units, which are DMPNet(MCSRN), DMPNet(CSRN) and DMPNet(CLSTM). In addition, to verify the effectiveness of error propagation unit in DMPNet, we also design a variation by deleting the error propagation unit in DMPNet(CLSTM) and call it DMPNet(W/O EP) for short.

For comparisons with recent approaches, we predict the next 3 frames given an input sequence of 3 frames and frame loss is computed by L2-based metrics since DFN[1] reports its results using mean squared error (MSE). Fig. 2 shows the quantitative comparison. It can be obviously seen that all the DMPNet variations outperform other two state-of-the-art methods. In addition, DMPNet(MCSRN) outperforms DMPNet(CSRN) and DMPNet(CLSTM). The reason is that the Highway Driving dataset only has small amount of data and the scene is relatively simple, which results that DMPNet does not need complicate memory unit to learn the contextual information. The performance of DMPNet is superior to DMPNet(W/O EP) which indicates that the prediction unit is useful and can improve the subsequent video prediction. As for the inference time, the above-mentioned methods, DMPNet, DFN and PredNet, are 0.01184s, 0.00609s, 0.00632s respectively on a TiTan X GPU. Although the computing cost of DMPNet is relatively larger, it is reasonable because it contains more neural cells to capture previous information.

4.3. Evaluation on KITTI

We next evaluate the DMPNet on a more complex video prediction dataset, KITTI [30], which is captured by a car-mounted camera on a car driving around an urban environment in Germany. The dataset includes city, residential and road categories. The training, validation and testing set consist of 41,396, 154 and 832 frames, respectively. Image frames are center-cropped and resized to 64 × 64 pixels. Our data preprocessing is same with [15] except for the image size. Considering the limited computer memory resources, all image frames are resized to 32 × 32 in this experiment.

Here, we conduct two groups of experiments in KITTI dataset. First is to predict the next one frame when given an input of 9 frames. This experiment is designed to compare with PredNet[2] fairly, because PredNet is good at one-step video prediction. As for the second task, multi-step video prediction, we generate the next 5 frames given an input of previous 5 frames. MSE and SSIM metrics are used as the measurements on these two experiments.

Tables 1 and 2 give the quantitative comparisons for one-step and multi-step predictions, respectively. It is obvious that DMPNet(CLSTM) outperforms all other methods. As we know, DMPNet(MCSRN) achieves the best performance on Highway Driving dataset, which is different from the experiment result on KITTI dataset. One main reason for this difference maybe that KITTI dataset is more complicate, and has 41K frames for training. It

[1] We use their released code at https://github.com/dbbert/dfn.

[2] We use their released code at https://github.com/coxlab/prednet.

Table 1
Quantitative comparison of different methods on KITTI dataset for one-step prediction.

	DFN	PredNet	DMPNet (MCSRN)	DMPNet (CSRN)	DMPNet (W/O EP)	**DMPNet (CLSTM)**
MSE	17.11	16.28	15.95	16.68	16.59	**15.37**
SSIM	0.84259	0.84028	0.84550	0.84550	0.83953	**0.85293**

Table 2
Quantitative comparison of different methods on KITTI dataset for multi-step prediction.

	DFN	PredNet	DMPNet (MCSRN)	DMPNet (CSRN)	DMPNet (W/O EP)	**DMPNet (CLSTM)**
MSE	42.66	74.71	39.75	40.62	39.76	**39.19**
SSIM	0.66272	0.55211	0.67715	0.66871	0.67639	**0.67985**

is nearly three times bigger than Highway Driving dataset. Meanwhile, DMPNet(CLSTM) contains more neural cells than DMPNet(MCSRNN). More training data can reduce over-fitting risk and make the more complicate memory unit, CLSTM, perform better. In addition, DMPNet(CLSTM) outperforms other two state-of-the-art methods, DFN and PredNet, in both one-step and multi-step prediction. These results convincingly show the effectiveness of our proposed DMPNet method. Fig. 3 presents qualitative visualized results of multi-step prediction by DMPNet(CLSTM), DFN and PredNet. In Fig. 3a, our DMPNet model is able to predict the process that the roof-mounted camera slowly approaches the truck and the truck becomes bigger and bigger. In Fig. 3b, DMPNet predicts the position of a tree, as the vehicle is running quickly on the road. In a word, DMPNet can better predict the dynamic process than DFN and PredNet, which is proved from both quantitative and qualitative experimental results.

5. Conclusion

This paper proposes a general deep learning neural network, DMPNet, for real-world video prediction. Actually, its design philosophy is inspired from the concept of memory mechanism and predictive coding from the cognitive neuroscience. DMPNet includes memory unit, prediction unit and an error propagation unit. To further investigate the memory unit, three different variations are designed and explored. The wide and convincing experiments are conducted on two public datasets. Experimental results demonstrate that memory and error propagation unit improve the quality of the pixel-level future prediction, and our proposed DMPNet model overall achieves state-of-the-art performance in predicting future frames on challenging real-world video datasets.

Acknowledgments

This work was partially supported by 973 Program under contract No 2015CB351802, Natural Science Foundation of China under contracts Nos. 61390511, 61472398, 61532018.

References

[1] N. Srivastava, E. Mansimov, R. Salakhudinov, Unsupervised learning of video representations using LSTMs, in: Proceedings of the International Conference on Machine Learning, 2015.
[2] Z. Luo, B. Peng, D.-A. Huang, A. Alahi, L. Fei-Fei, Unsupervised learning of long-term motion dynamics for videos, in: Proceedings of the International Conference on Computer Vision and Pattern Recognition, 2017.
[3] V. Patraucean, A. Handa, R. Cipolla, Spatio-temporal video autoencoder with differentiable memory, arXiv:1511.06309 (2015).
[4] N. Sedaghat, Next-flow: Hybrid multi-tasking with next-frame prediction to boost optical-flow estimation in the wild, arXiv:1612.03777 (2016).
[5] C. Vondrick, H. Pirsiavash, A. Torralba, Anticipating the future by watching unlabeled video, arXiv:1504.08023 (2015).
[6] T. Lan, T.-C. Chen, S. Savarese, A hierarchical representation for future action prediction, in: Proceedings of the European Conference on Computer Vision, 2014.
[7] J. Yuen, A. Torralba, A data-driven approach for event prediction, in: Proceedings of the European Conference on Computer Vision, 2010.
[8] M. Ranzato, A. Szlam, J. Bruna, M. Mathieu, R. Collobert, S. Chopra, Video (language) modeling: a baseline for generative models of natural videos, arXiv:1412.6604 (2014).
[9] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-c. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in: Proceedings of the Advances in Neural Information Processing Systems, 2015.
[10] J. Oh, X. Guo, H. Lee, R.L. Lewis, S. Singh, Action-conditional video prediction using deep networks in Atari games, in: Proceedings of the Advances in Neural Information Processing Systems, 2015.
[11] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, H. Lee, Learning to generate long-term future via hierarchical prediction, arXiv:1704.05831(2017).
[12] N. Kalchbrenner, A.v. d. Oord, K. Simonyan, I. Danihelka, O. Vinyals, A. Graves, K. Kavukcuoglu, Video pixel networks, arXiv:1610.00527 (2016).
[13] B. De Brabandere, X. Jia, T. Tuytelaars, L. Van Gool, Dynamic filter networks, in: Proceedings of the Advances In Neural Information Processing Systems, 2016.
[14] C. Finn, I. Goodfellow, S. Levine, Unsupervised learning for physical interaction through video prediction, in: Proceedings of the Advances In Neural Information Processing Systems, 2016.
[15] W. Lotter, G. Kreiman, D. Cox, Deep predictive coding networks for video prediction and unsupervised learning, in: Proceedings of the International Conference on Learning Representations, 2017.
[16] M.C. Anderson, B.A. Spellman, On the status of inhibitory mechanisms in cognition: memory retrieval as a model case., Psychol. Rev. 102 (1995) 68.
[17] L.L. Jacoby, L.R. Brooks, Nonanalytic cognition: memory, perception, and concept learning, Psychol. Learn. Motiv. 18 (1984) 1–47.
[18] J.T. Richardson, Working Memory and Human Cognition, 3, Oxford University Press on Demand, 1996.
[19] A.D. Baddeley, G. Hitch, Working memory, Psychol. Learn. Motiv. 8 (1974) 47–89.
[20] R.S. Fernández, M.M. Boccia, M.E. Pedreira, The fate of memory: reconsolidation and the case of prediction error, Neurosci. Biobehav. Rev. 68 (2016) 423–441.
[21] J. Hawkins, D. George, J. Niemasik, Sequence memory for prediction, inference and behaviour, Philos. Trans. Biol. Sci. 364 (2009) 1203.
[22] R.P. Rao, D.H. Ballard, Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects, Nat. Neurosci. 2 (1999) 79–87.
[23] S.J. Luck, E.K. Vogel, The capacity of visual working memory for features and conjunctions, Nature 390 (1997) 279–281.
[24] A. Baddeley, Working memory: looking back and looking forward, Nat. Rev. Neurosci. 4 (2003) 829–839.
[25] J.D. Cohen, W.M. Perlstein, T.S. Braver, L.E. Nystrom, D.C. Noll, J. Jonides, E.E. Smith, Temporal dynamics of brain activation during a working memory task, Nature 386 (1997) 604.
[26] Y. Bengio, How auto-encoders could provide credit assignment in deep networks via target propagation, arXiv:1407.7906 (2014).
[27] A. Graves, A.-r. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, in: Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 2013.
[28] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, in: Proceedings of the Advances in neural information processing systems, 2014.
[29] M. Liwicki, A. Graves, H. Bunke, J. Schmidhuber, A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks, in: Proceedings of the International Conference on Document Analysis and Recognition, 2007.
[30] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: the Kitti dataset, Int. J. Robot. Res. 32 (2013) 1231–1237.
[31] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the International Conference on Computer Vision, 2015.

[32] D. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980 (2014).

[33] PyTorch, https://github.com/pytorch/pytorch (2017).

Zhipeng Liu received the B.S. degree in computer science from the Harbin Engineering University, Harbin, China, in 2011. He is currently working on his M.S. degree in computer science at the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China. His research interests include computer vision and pattern recognition, especially sign language recognition.

Xiujuan Chai received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2000, 2002, and 2007, respectively. She was a Post-doctorial researcher in Nokia Research Center (Beijing), from 2007 to 2009. She joined the Institute of Computing Technology, Chinese Academy Sciences, Beijing, in July 2009 and now she is an Associate Professor in Agricultural Information Institute, Chinese Academy of Agricultural Sciences. Her research interests cover computer vision, pattern recognition, and multimodal human-computer interaction. She especially focuses on sign language recognition related research topics.

Xilin Chen is a professor of ICT, CAS. He has authored one book and more than 200 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He served as an Organizing Committee/Program Committee member for more than 70 conferences. He was a recipient of several awards, including the China's State Natural Science Award in 2015, the China's State S&T Progress Award in 2000, 2003, 2005, and 2012 for his research work. He is currently an associate editor of the IEEE Transactions on Multimedia, a leading editor of the Journal of Computer Science and Technology, and an associate editor-in-chief of the Chinese Journal of Computers. He is a fellow of the China Computer Federation (CCF), IAPR, and the IEEE.