Motion-Aware Feature Enhancement Network for Video Prediction

Xue Lin, Qi Zou, Xixia Xu, Yaping Huang, and Yi Tian

***Abstract*— Video prediction is challenging, due to the pixel-level precision requirement and the difficulty in capturing scene dynamics. Most approaches tackle the problems by pixel-level reconstruction objectives and two decomposed branches, which still suffer from blurry generations or dramatic degradations in long-term prediction. In this paper, we propose a Motion-Aware Feature Enhancement (MAFE) network for video prediction to produce realistic future frames and achieve relatively long-term predictions. First, a Channel-wise and Spatial Attention (CSA) module is designed to extract motion-aware features, which enhances the contribution of important motion details during encoding, and subsequently improves the discriminability of attention map for the frame refinement. Second, a Motion Perceptual Loss (MPL) is proposed to guide the learning of temporal cues, which benefits to robust long-term video prediction. Extensive experiments on three human activity video datasets: KTH, Human3.6M, and PennAction demonstrate the effectiveness of the proposed video prediction model compared with the state-of-the-art approaches.**

***Index Terms*— Video prediction, unsupervised learning, attention mechanism, perceptual loss.**

I. Introduction

UNDERSTANDING the dynamics of scenes and predicting future states is one of the core problems in computer vision and artificial intelligence. A good predictor should have an accurate internal representation of the video evolution, and therefore benefits some video-understanding-related tasks such as action recognition [1]–[3], activity anticipation [4], [5] and interaction inference [6]. It also has great potential in autonomous driving [7] and task planning [8], [9].

However, generating realistic videos with consistent dynamics from raw frames is challenging, due to high dimensionality of raw feature space and complex motions and interactions. This is more difficult when no supervision information is available. Some methods try to leverage Long-Short Term Memory (LSTM) networks to learn dynamics from videos. To generate realistic future frames, some methods extend techniques used for image synthesis to video prediction, such as Generative Adversarial Networks (GANs). In this paper, we point out two factors which are important but neglected in different degrees.

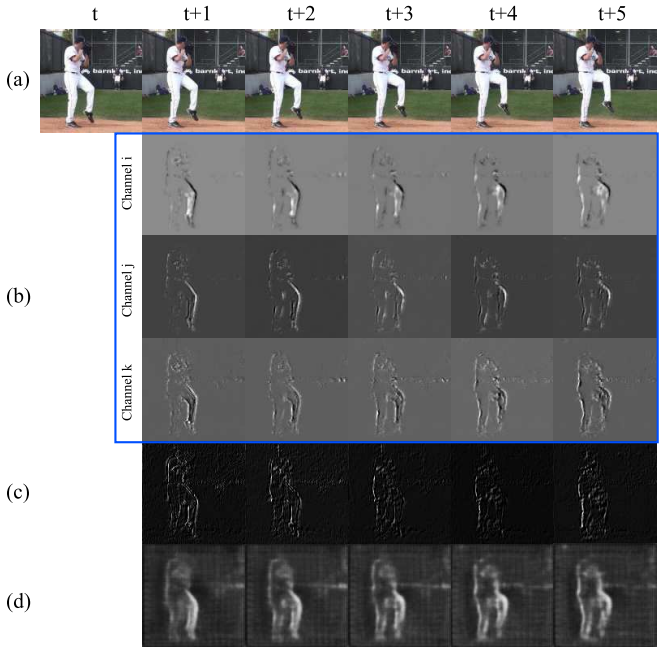


Fig. 1. The dynamic focuses of attention over time. (a) The consecutive video sequence frames. (b) Different feature maps computed by our MAFE module. (c) The feature map computed by MCnet [10]. (d) The attention map computed by us. The different degrees of highlight areas correspond to the different levels of attention. The brighter the area is, the more important it is.

A. Key Information Should be Adaptively Selected From Moving Objects

Many works disentangle moving objects from static background for video prediction, while do not further distinguish different parts in a moving object. However, different parts are not equally important for learning the dynamics. As shown in Fig. 1, the dynamics of the person pitching baseball can be mainly represented by the motion patterns of one of his legs and feet. Intensive attention on these parts will help to produce realistic future frames. Different levels of attention should be adaptively assigned to different parts, and such attention varies across time. Based on the above observations, we propose a channel-wise and spatial attention to selectively emphasize informative features, which can further enhance the discriminative capacity of attention map to focus more on the main part of moving objects.

Manuscript received November 22, 2019; revised March 1, 2020; accepted March 30, 2020. Date of publication April 10, 2020; date of current version February 4, 2021. This work was supported by the Fundamental Research Funds for the Central Universities under Grant 2019YJS030. This article was recommended by Associate Editor Y. Tan. *(Corresponding author: Qi Zou.)*

The authors are with the Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China (e-mail: 18112028@bjtu.edu.cn; qzou@bjtu.edu.cn; 18120432@bjtu.edu.cn; yphuang@bjtu.edu.cn; tianyi@bjtu.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2020.2987141

B. High-Level Motion Representation is Essential for Future Prediction

Many approaches exploit pixel-level loss for video prediction, such as pixel-level regression loss [2], [6], [11]–[13] and adversarial loss [10], [14], [15]. However, low-level information varies dramatically over time. It is common that predicted frames are of high quality for the first few steps while degrade greatly through time. Studies in action recognition [16]–[20] suggest that high-level motion features are the major factors for explaining the pixel variations into the future, and thus are more stable and essential for temporal dynamics representation. Therefore, we are motivated to explicitly use the high-level encoding of motion information as an important signal for learning robust long-term video prediction.

In this work, we propose a novel video prediction method named **M**otion-**A**ware **F**eature **E**nhancement network (**MAFE**). The proposed framework contains three modules: frame encoder, motion predictor, and frame generator. Exploring the temporal motion cues is of great importance for video generation, so our framework processes motion and content information independently. In consideration of the different characteristics of different level features, for low-level features, we design a **C**hannel-wise and **S**patial **A**ttention (**CSA**) module integrated into the low-layer of motion encoder to selectively emphasize informative motion features as well as suppress useless ones. The enhanced motion-aware features greatly contribute to the discriminative representations of attention map for different parts of moving objects. Our proposed CSA can easily complement other video prediction models for learning the motion-sensitive features to further boost their performances. For high-level features, inspired by action recognition [16]–[20] as well as perceptual loss of style transfer [21], we propose a **M**otion **P**erceptual **L**oss (**MPL**) based on the high-level motion encoding to reduce the dramatic degradation of video prediction quality over time, which is common in methods with pixel-level losses. The proposed MPL minimizes the motion perceptual features obtained from motion encoder and motion predictor, which enforces the model to learn the dynamic motion patterns and achieve relatively long-term video prediction. The frame generator contains a content decoder as well as additional refinement with the aforementioned attention map, which can effectively promote the model to produce more realistic future frames.

To summarize, our contributions are as follows:

(1) We design a **C**hannel-wise and **S**patial **A**ttention (**CSA**) module to enhance motion-aware features, which can significantly improve the discriminative capability of attention map for different parts of moving objects.

(2) We propose a novel **M**otion **P**erceptual **L**oss (**MPL**) based on high-level motion features to guide the model to learn more about the temporal cues and achieve robust long-term video prediction.

(3) We perform extensive experiments on the KTH, Human3.6M, and PennAction datasets to validate the effectiveness of our proposed model, and show that it can achieve state-of-the-art results on these benchmarks.

The remainder of this paper is organized as follows: Section II describes some works related to our work. In Section III, we describe our proposed approach. In Section IV, we demonstrate the performance of our approach by comparing it with the state-of-the-art algorithms. The paper concludes with a discussion of our work in Section V.

II. Related Works

A. Video Prediction

B. Supervised Prediction

Semantic segmentation masks as supervised signals. Wang *et al.* [22] propose to synthesize a photorealistic video from a source video, which achieves high-resolution, photorealistic and temporally coherent video results on a diverse set of input formats. However, it needs a sequence of semantic segmentation masks as inputs that are hard to obtain. Pan *et al.* [23] further relax the conditions, that is, to use a single semantic label map as input to generate videos. The generation given semantic labels can obviously help avoid undesirable results which often occur in unconditional generation.

High-level features as supervised signals. Jang *et al.* [24] propose to use appearance and motion information as conditions to train a model based on GAN. Villegas *et al.* [15] propose a hierarchical method for making long-term predictions. It first inputs the human pose, treated as high-level structure, into the ConvLSTM to predict the future human pose and then generates the future frames by visual-structure analogy. We also utilize high-level features as supervised information, but in contrast to the above works, we do not need any annotations to learn the high-level features.

C. Unsupervised Prediction

The unsupervised video prediction methods are mainly categorized as deterministic prediction and stochastic variational prediction. In addition, the two-stream approach is also an important technique for video prediction.

Deterministic video prediction. A range of deep video models have recently been proposed. Srivastava *et al.* [3] propose Long Short-Term Memory (LSTM) [25] based on encoder-decoder models for the task of video prediction. However, the LSTM does not consider the spatial structure of video data. This inspires the development of Convolutional LSTM (ConvLSTM) [26] which replaces the internal transformations of an LSTM cell with convolutions. Finn *et al.* [6] train the models based on ConvLSTM to predict pixel motions rather than values by predicting a distribution over pixel motion from previous frames. Lotter *et al.* [2] develop the deep predictive neural network (PredNet) to predict future frames, in which each layer makes local predictions and only forwards deviations to subsequent network layers based on ConvLSTM units. Byeon *et al.* [12] identify the important factor for imprecise prediction of the previous methods, such as blind spots of the ConvLSTM. Thus, they introduce a fully context-aware architecture (ContextVP) to capture the entire available past context. Xu *et al.* [11] propose a RNN structure for video prediction that employs temporal-adaptive convolutional kernels to capture time-varying motion patterns and tiny objects.

Wang *et al.* [27] propose Eidetic 3D LSTM (E3D-LSTM) for video prediction that can recall the long-range historical context by learning to attend to previous memory states via a gate-controlled self-attention module. However, the self-attention mechanism needs to recall temporally distant memory and the memory state of E3D-LSTM is learned to attend to all previous relevant moments, which is complicate and hard to train. Wichers *et al.* [14] develop hierarchical long-term video prediction without requiring ground truth high-level structure annotations, which encodes the input frame and predicts a high-level encoding into the future. Our component MPL is inspired by the high-level encoding introduced by [14]. However, our approach differs from these in that we predict high-level motion encoding rather than the high-level static content encoding.

Stochastic variational video prediction. Some researchers [28]–[31] consider the video prediction as a stochastic process. Emily Denton [29] introduce an unsupervised video prediction model that can learn a prior model of uncertainty in a given environment. Babaeizadeh *et al.* [30] develop a stochastic variational video prediction (SV2P) method that predicts a different possible future for each sample of its latent variables. Xu *et al.* [28] propose a two-stage framework called VPSS, which approaches the video prediction as a stochastic process in a new view. Lee *et al.* [31] design a model that combines both adversarial losses and latent variables to enable high-quality stochastic video prediction. Different from the deterministic prediction model, the stochastic variational model aims to produce diverse predictions that cover a range of possible futures. To understand whether the true future is within the set of predictions, most of stochastic variational methods usually evaluate the score of the best sample from multiple random priors, which is unpractical in reality and a little unfair for comparison with the deterministic methods.

Two-stream video prediction. Some researchers propose to separate the video into dynamic and static areas to deal with. For videos with static backgrounds and moving foreground, some researchers [32], [33] explicitly model the scenes moving foreground object separately from the background. Liu *et al.* [1] develop the architecture that first estimates the optical flow and applies it to generate the future frames. Villegas *et al.* [10] propose a motion-content network (MCnet), which splits the inputs into motion and content branches and independently captures different information with separate encoder pathways. However, MCnet [10] cannot capture the motion information in the long-term prediction. Our proposed MPL use the high-level motion encoding as supervision to guide the model to learn about the temporal cues. Note that our model learns high-level motion features in an unsupervised way, i.e., without any ground truth annotations. What is more, MCnet is easy to encounter the loss of structural details, which is due to the fact that the encoder of MCnet deals with the features equally and lacks discriminative learning ability. Our proposed channel-wise and spatial attention module in the motion encoder can emphasize the informative features related to the video prediction to restore the detailed structures.

D. Perceptual Optimization

A large number of papers have used perceptual optimization to generate images, depending on high-level features extracted from a convolutional network. The objective can be maximizing class prediction scores [34] or individual features [34] to understand the functions encoded in trained models. The similar methods can also be applied to crafting adversarial examples [35], [36]. Additionally, Mahendran and Vedaldi [37] explore to invert features from convolutional networks by minimizing a feature reconstruction loss to understand the image information retained by different network layers. Some previous researchers use the methods to invert local binary descriptors [38] and HOG features [39]. This strategy has been applied to texture synthesis and style transfer by Gatys *et al.* [40], [41]. Some works learn discriminative motion representation for action recognition by perceptual optimization of high-level motion-related features. Crasto *et al.* [18] distills knowledge from the flow to the RGB stream by matching high-level features. Shou *et al.* [19] designs a generator network that can learn to predict discriminative motion cues by using optical flow as supervision and being trained jointly with action classifier. Ng *et al.* [20] considers the optical flow learning as an auxiliary task to force the model effectively to learn useful representations from motion modeling. Inspired by these, we propose a motion perceptual loss based on the high-level motion encoding features in order to learn more the temporal cues for long-term video prediction. The difference between the perceptual optimization for action recognition and our MPL is that our proposed motion perceptual loss is self-supervised, without any labelled supervised information.

E. Attention Mechanism

The goal of attention mechanism in deep neural network is to recalibrate the feature responses towards the most informative and important components of the inputs. The attention mechanism has been successfully applied in various tasks such as action recognition [42], image generation [43], image captioning [44], [45], image classification [46], [47], object recognition [48], saliency detection [49], image super-resolution [50] and pose estimation [51]. Xu *et al.* [45] propose a visual attention model for image captioning, which uses hard pooling to select the most probably attentive regions to average the spatial features with attentive weights. By exploring the interdependencies between the channels of deep features, Hu *et al.* [46] propose a squeeze-and-excitation (SE) block to adaptively recalibrate channel-wise feature responses for image classification.

To capture the accurate and subtle differences of different parts in their contributions to video prediction, we combine channel-wise and spatial attentions into the motion encoder to adaptively focus on the main motion part.

III. Proposed Method

In this paper, we propose a novel video prediction method named **M**otion-**A**ware **F**eature **E**nhancement network (**MAFE**), which contains three modules: frame encoder with

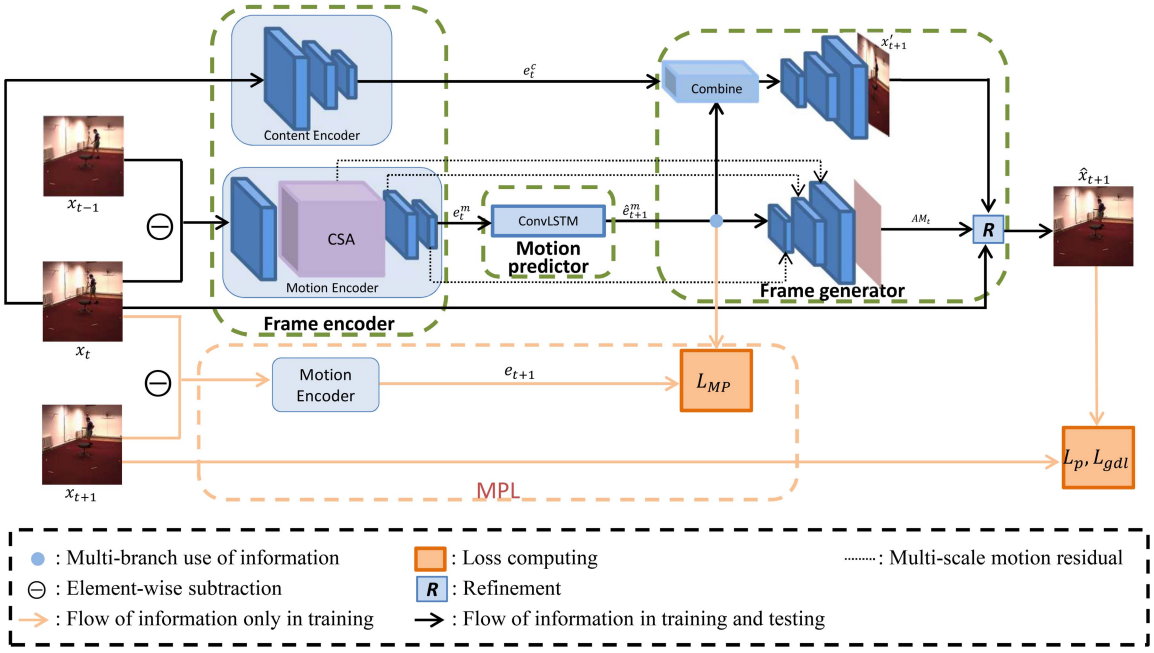


Fig. 2. The overall architecture of our proposed **MAFE** network. The frame encoder (Section III-A) takes two consecutive frames as inputs and outputs the high-level content features e_t^c and enhanced motion-ware features e_t^m. The high-level motion features e_t^m are fed into the motion predictor (Section III-B) for producing the next motion encoding $\hat{e}_{t+1}^m$. The frame generator (Section III-C) takes e_t^c and $\hat{e}_{t+1}^m$ as inputs and generates the final refined prediction. Besides the above losses, another adversarial loss (Section III-D) is included to enforce the model to generate realistic looking images. In addition, a discriminator network (Section III-D) is designed to distinguish the predicted future frames from the truth frames.

Channel-wise and **S**patial **A**ttention (**CSA**), motion predictor with **M**otion **P**erceptual **L**oss (**MPL**) and frame generator with soft attention map. In the encoder described in section III-A, we split the inputs into two streams, motion and content, and independently capture different information with motion encoder and content encoder. The CSA module is applied to the low-level motion space for emphasizing the informative motion features as well as suppressing useless ones, which can implicitly improve the discriminative ability of the soft attention map for frame generator. In the motion predictor component described in section III-B, the MPL is designed to enforce the model to learn the long-term motion patterns by using the high-level features as supervision. In the generator shown in section III-C, the soft attention map is computed to adaptively focus on the dynamic areas and copy some pixels from the previous frame to avoid the loss of static structures. The overall architecture is illustrated in Fig. 2.

Let $x_t \in \mathbb{R}^{W\times H\times C}$ denote the t-th frame in a video x, where W, H and C stand for width, height, and number of channels, respectively. Supposing that the network observes K frames and predicts the next T frames in the training or testing process. It recursively performs the above prediction over T time steps. That is to say, our network observes a history of consecutive frames $\{1, 2, ..., K\}$ to generate the next frame $K+1$, and the predicted frame $K+1$ joins the previous sequence to form the new sequence $\{1, 2, ..., K, K+1\}$ for the next prediction $K+2$. Repeat it until all T frames are generated.

A. Frame Encoding

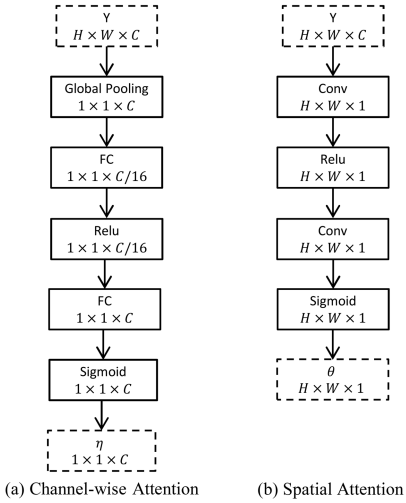
The frame encoder contains motion encoder and content encoder that respectively deal with the dynamics of consecutive frames and the spatial layout. For video prediction, extracting the effective motion patterns is of great importance due to the fact that the high-level motion features will be considered as the key information for the predictor. In order to exploit the motion pattern of consecutive frames, we design a channel-wise and spatial attention module integrated into the motion encoder to adaptively recalibrate the feature responses both in the channel-wise and spatial context.

*1) Motion Encoder Based on Channel-Wise and Spatial Attention (**CSA**):* In our model, the motion encoder is fed with the results of frame difference x_{t-1} and x_t, and outputs the high-level encoding of motion trend as follows.

$$\begin{aligned} e_t^m &= \Phi_m(x_t - x_{t-1}, \Theta_m),\ 2 \le t \le K, \\ \hat{e}_t^m &= \Phi_m(\hat{x}_t - \hat{x}_{t-1}, \Theta_m),\ K+1 \le t \le (K+T), \end{aligned} \quad (1)$$

where $x_t - x_{t-1}$ represents the element-wise subtraction, e_t^m is the high-level motion perceptual features got from the frames x_t and x_{t-1}. Φ_m is the motion encoder with parameters Θ_m whose detailed architecture can be found in section IV.

Generally, the motion encodings contain different types of information in the channel-wise and spatial context which have different contributions to the video prediction. In order to enhance the representational power of the network and prompt it focus on the key information of moving objects, we design a **C**hannel-wise **A**ttention (**CA**) unit and a **S**patial **A**ttention (**SA**) unit as shown in Fig. 3, and then combine two types of attention into the motion encoder to augment the feature representations as shown in Fig. 4. Following, we will describe the channel-wise attention unit and spatial attention unit in more detail.



(a) Channel-wise Attention (b) Spatial Attention

Fig. 3. The diagram of channel-wise and spatial attention units. (a) The operations of **C**hannel-wise **A**ttention (**CA**) unit, by which the channel-wise attention weights are calculated. (b) The operations of **S**patial **A**ttention (**SA**) unit, by which the spatial attention weights are computed.

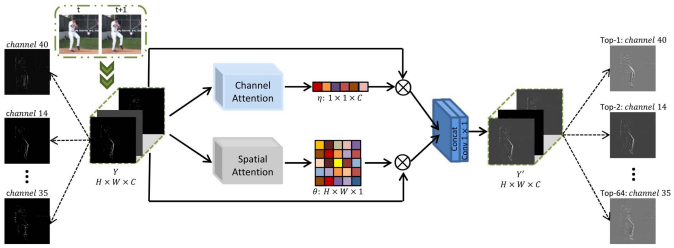


Fig. 4. The **C**hannel-wise and **S**patial **A**ttention (**CSA**) module integrated into the low-layer of motion encoder. The CSA module takes Y as input and outputs the calibrated feature map $Y^{'}$. The motion-related channels, such as *channel* 40 and *channel* 14, are assigned higher weights. For a channel, such as *channel* 40, the motion-sensitive positions are emphasized.

Channel-wise Attention (CA). Generally, different channels of CNN features express diverse semantic information [46]. The channel-wise attention that can emphasize informative feature maps while suppress useless ones will exactly concentrate on the main part of motion.

Assuming the input of the channel-wise attention is $Y \in \mathbb{R}^{H\times W\times C}$ that represents the feature maps obtained from first block of the motion encoder, the output is weights $\eta \in \mathbb{R}^{1\times 1\times C}$. Then we can perform feature recalibration by $U \in \mathbb{R}^{H\times W\times C} = \eta \otimes Y$, where $\otimes$ is a element-wise multiplication in the channel-wise. Inspired by SE-Net [46], channel-wise attention weights can be obtained by two steps: squeeze step and excitation step. In the squeeze step, global average pooling is operated to calculate the channel-wise statistics $z \in \mathbb{R}^{1\times 1\times C}$ of input Y. The c-th element of z is calculated by

$$z_c = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W} y_c(i, j), \tag{2}$$

where $y_c(i, j)$ is the value at position (i, j) of the c-th channel y_c. In the excitation step, a simple gating mechanism with a sigmoid activation is employed to summary statistics z. The process is represented as follows.

$$\eta = \sigma(W^2_{CA} * \delta(W^1_{CA} * z + b^1_{CA}) + b^2_{CA}), \tag{3}$$

where $\sigma(\cdot)$ and $\delta(\cdot)$ represent the sigmoid and ReLU functions respectively, and $*$ stands for the convolution operator. $W^1_{CA} \in \mathbb{R}^{1\times 1\times \frac{C}{r}\times C}$, $b^1_{CA} \in \mathbb{R}^{1\times 1\times \frac{C}{r}}$, $W^2_{CA} \in \mathbb{R}^{1\times 1\times C\times \frac{C}{r}}$, and $b^2_{CA} \in \mathbb{R}^{1\times 1\times C}$ are the parameters of the two fully connected layers.

With the above process, the CA unit is able to adaptively modulate the channel-wise features according to the channel-wise statistics of input, and helps the network boost the channel-wise feature discriminability.

Spatial Attention (SA). The video can be decomposed into moving objects and static background. For moving objects, different parts in a moving object are not equally important for learning the dynamics as shown in Fig. 1. Therefore, instead of considering all spatial position equally, we adopt spatial attention to focus more on the dynamic areas.

Assuming the input of spatial attention is $Y \in \mathbb{R}^{H\times W\times C}$, the output is weights $\theta \in \mathbb{R}^{H\times W\times 1}$. Then we can weight $Y \in \mathbb{R}^{H\times W\times C}$ with θ to get the calibrated features $V \in \mathbb{R}^{H\times W\times C}$, i.e., $V = \theta \otimes Y$, where $\otimes$ is a element-wise multiplication in spatial context. The spatial attention weights are generated by two convolutional operations.

$$\theta = \sigma(W^2_{SA} * \delta(W^1_{SA} * Y + b^1_{SA}) + b^2_{SA}), \tag{4}$$

where $\sigma(\cdot)$, $\delta(\cdot)$ and $*$ have the same meanings as those used in Eq. (3). $W^1_{SA} \in \mathbb{R}^{1\times 1\times C}$, $b^1_{CA} \in \mathbb{R}^{H\times W\times 1}$, $W^2_{SA} \in \mathbb{R}^{1\times 1\times 1}$, and $b^2_{SA} \in \mathbb{R}^{H\times W\times 1}$ denote the parameters of two fully connected layers.

Integration of CA and SA into motion encoder. The features obtained from CA and SA emphasize the importance of different semantics and positions of moving object respectively. The fusion of these features enhances the ability to express motion-aware features. For this, we empirically concatenate U and V and then input to a 1×1 convolutional layer as follows.

$$Y^{'} = W_\phi * [U, V] + b_\phi, \tag{5}$$

where W_ϕ and b_ϕ are the parameters of convolutional layer, $[\cdot]$ represents the operation of feature concatenation. Specially, as shown in Fig. 4, the motion-related feature maps, such as $Top - 1$ and $Top - 2$ channel, can be selected and emphasized by CA. For a feature map, such as *channel* 40, the motion-aware positions are enhanced compared with the ones without SA.

The calibrated features $Y^{'} \in H \times W \times C$ are fed into the next layer of motion encoder to obtain the final high-level features.

2) Content Encoder: As a counterpart to the motion encoder, the content encoder deals with static scene information like spatial layout. In our work, the last observed frame is used to produce the content features for the future prediction. The high-level content features e^c_t can be obtained by content encoder defined by

$$e^c_t = \Phi_c(x_t, \Theta_c),\ 1 \le t \le (K + T), \tag{6}$$

where Φ_c is content encoder with parameters Θ_c that specializes on extracting features from a single frame similar

to MCnet [10]. The detailed architecture of Φ_c is shown in section IV.

B. Motion Prediction of High-Level Features

The high-level encoding of motion cues is stable and essential for temporal dynamic representations in future prediction. We use the Convolutional LSTM as a motion predictor to produce high-level motion cues for robust long-term video prediction.

1) Motion Predictor: The motion perceptual features obtained by motion encoder contain the local dynamics of the two consecutive frames. In order to obtain the complicated intrinsic temporal features in a longer time span, we use a ConvLSTM layer on top defined by

$$\begin{aligned}[C_t, H_t] &= ConvLSTM(e_t^m, C_{t-1}, H_{t-1}) \quad if\ t \le K,\\ [C_t, H_t] &= ConvLSTM(\hat{e}_t^m, C_{t-1}, H_{t-1}) \quad if\ t > K, \end{aligned} \tag{7}$$

where C_t is the memory cell that retains information from the history of the inputs, H_t represents the hidden state of the ConvLSTM at time step t, K represents the number of observed frames. For the observed frames, the input of ConvLSTM e_t^m is obtained from motion encoder. For the predicted frames, the input $\hat{e}_t^m$ is computed by the ConvLSTM. Generally, the hidden state H_t can approximately express the prediction at time $t+1$, i.e.,

$$\hat{e}_{t+1}^m = H_t. \tag{8}$$

The key equations of $ConvLSTM(e_t^m, C_{t-1}, H_{t-1})$ are shown as follows.

$$\begin{aligned} i_t &= \sigma(W_{e^m i} * e_t^m + W_{hi} * H_{t-1} + W_{ci} \circ C_{t-1} + b_i),\\ f_t &= \sigma(W_{e^m f} * e_t^m + W_{hf} * H_{t-1} + W_{cf} \circ C_{t-1} + b_f),\\ C_t &= f_t \circ C_{t-1} + i_t \circ tanh(W_{e^m c} * e_t^m + W_{hc} * H_{t-1} + b_c),\\ o_t &= \sigma(W_{e^m o} * e_t^m + W_{ho} * H_{t-1} + W_{co} \circ C_{t-1} + b_o),\\ H_t &= o_t \circ tanh(C_t), \end{aligned} \tag{9}$$

where $W = \{W_{e^m i}, W_{hi}, W_{ci}, W_{e^m f}, W_{hf}, W_{cf}, W_{e^m c}, W_{hc}, W_{e^m o}, W_{ho}, W_{co}\}$ and $b = \{b_i, b_f, b_c, b_o\}$ denote the parameters of ConvLSTM. $*$ stands for the convolution operator and $\circ$ denotes the Hadamard product.

*2) **M**otion **P**erceptual **L**oss (**MPL**):* $\hat{e}_t^m$ predicted by ConvLSTM contains high-level motion information. But it is not explored in the phase of observing frames and only as the input of next time ConvLSTM in the phase of prediction. In order to fully exploit the high-level motion perceptual features, we propose a novel **M**otion **P**erceptual **L**oss (MPL), which uses the motion perceptual features as a supervised signal. In other words, the network is trained to constrain the features predicted by ConvLSTM to be close to the outputs of the motion encoder. It can be formulated as

$$L_{MP} = \sum_{t=3}^{K+T} \|e_t^m - \hat{e}_t^m\|_\varsigma^\varsigma, \tag{10}$$

where ς is hyperparameter that decides which loss will be used. That is to say, $\|\cdot\|_1^1$ means ℓ_1 loss and $\|\cdot\|_2^2$ means ℓ_2 loss. The time range is from 3 to $K+T$ because of the output of motion encoder $e_t^m = \{e_2^m, e_3^m, ..., e_{K+T}^m\}$ and the prediction of the ConvLSTM $\hat{e}_t^m = \{e_3^m, e_4^m, ..., e_{K+T}^m\}$. It should be noticed that the proposed motion perceptual loss can be easily integrated into other methods based on recurrent architectures in order to improve the learning capacity of models.

C. Frame Generation

The frame generator contains two streams, content decoder and refinement with the soft attention map. The content decoder is fed with high-level content encoding features and outputs the next frame roughly. However, most of the pixel areas in video are near copies of areas in nearby existing frames. Totally generating the future frame from scratch is harder than copying some static pixels from previous frame [1]. Therefore, we compute a soft attention map to automatically weigh the frame obtained from the content decoder stream and the previous frame to refine the prediction.

1) Content Decoder: The high-level content encoding e_t^c of the last observed frame is more informative about the future spatial layout. We transform e_t^c using the dynamics feature encoding in $\hat{e}_{t+1}^m$ to generate content feature of the next time step e_{t+1}^c and feed e_{t+1}^c to a content decoder Ψ_c to produce the original pixel space $x_{t+1}^{'}$. It can be formulated as follows.

$$x_{t+1}^{'} = \Psi_c(\Gamma(\hat{e}_{t+1}^m, e_t^c), r_t^c, \Lambda_c),\ 2 \le t \le (K+T-1), \tag{11}$$

where $\Gamma(\hat{e}_{t+1}^m, e_t^c)$ denotes a combined module that is implemented by a CNN with bottleneck layers parameterized by Λ_c to obtain the content features of next time step e_{t+1}^c. r_t^c is a list containing the residual connection from every layer of the motion and content encoder before pooling. It is sent to every layer of the content decoder after unpooling. The content decoder is composed of multiple successive operations of deconvolution, rectification and unpooling. The output layer is passed through a $tanh(\cdot)$ activation function [10].

2) Refinement With the Soft Attention Map: For video prediction, the key information should be adaptively selected from moving objects to produce realistic future frames. Moreover, the enhanced motion-aware features obtained from motion encoder emphasize the features that greatly contribute to the dynamics of moving objects. The above two observations inspire us to compute a soft attention map based on the motion-aware features to adaptively focus on the dynamic areas. Thus, we use the soft attention map to refine the predicted frames, which can avoid the loss of static area to some extent.

The soft attention map is obtained from a soft attention block Ψ_{AM}. The Ψ_{AM} consists of several simple convolutional layers that share weights with the content decoder Ψ_c and a final sigmoid transformation layer mapped to [0, 1]. It can be formulated as follows.

$$AM_t = \Psi_{AM}(\hat{e}_{t+1}^m, r_t^m, \Lambda_{AM}), \quad t > K, \tag{12}$$

where $AM_t \in \mathbb{R}^{H \times W \times C}$ stands for the soft attention map at time t, Λ_{AM} is the parameters of Ψ_{AM}, and $t > K$ because we only compute the attention map to generate the future in the process of prediction. Similar to r_t^c, r_t^m is a list containing the residual connection, but differently it is from each layer of the

motion encoder before pooling. It is sent to every layer of the soft attention block Ψ_{AM} after unpooling. The soft attention map is then used to guide the model for the final predicted frame as follows.

$$\hat{x}_{t+1} = AM_t \otimes x^{'}_{t+1} + (1 - AM_t) \otimes x_t, \tag{13}$$

where $\otimes$ represents element-wise product operator and 1 is an image of all ones. The first part produces the next new pixels, while the second part corresponds to pixels warped from the previous frame. Our soft attention map can effectively identify the key areas of moving objects in the video frames and keep the static area perfectly.

D. Implementation Details

To train our network, we use the compound loss similar to [10]. Differently, we add motion perceptual loss L_{MP} as a supervised signal. Our network is optimized to minimize the objective given by

$$L = \alpha L_{img} + \beta L_{GAN} + \gamma L_{MP}, \tag{14}$$

where α, β and γ are hyper-parameters that control the effect of each sub-loss during the training. γ is hyper-parameters that controls the effect of motion perceptual loss mentioned in section III-B.2. L_{img} is the loss in the image space defined as follows.

$$L_{img} = L_p + L_{gdl}, \tag{15}$$

$$L_p(x_{t+1}, \hat{x}_{t+1}) = \|x_{t+1} - \hat{x}_{t+1}\|_2^2 \tag{16}$$

and

$$\begin{aligned} L_{gdl}(x_{t+1}, \hat{x}_{t+1}) = \sum_{i,j}^{h,w} &|(|x^{i,j}_{t+1} - x^{i-1,j}_{t+1}| - |\hat{x}^{i,j}_{t+1} - \hat{x}^{i-1,j}_{t+1}|)| \\ &+ |(|x^{i,j-1}_{t+1} - x^{i,j}_{t+1}| - |\hat{x}^{i,j-1}_{t+1} - \hat{x}^{i,j}_{t+1}|)|. \end{aligned} \tag{17}$$

Combining L_p and L_{gdl} guides our network to match the average pixel values directly while also matching the gradients of such pixel values. However, they only consider the reconstruction loss of a static frame rather than the two consecutive frames. Our proposed loss L_{MP} can effectively capture local dynamics of the two consecutive frames. In addition, we use an adversarial training strategy to make our model generate realistic looking. L_{GAN} is the term in adversarial loss that allows our model to generate realistic looking images and is defined by

$$L_{GAN} = -log D([x_{1:t}, G(x_{1:t})]), \tag{18}$$

where $x_{1:t}$ is the concatenation of the input frames along the depth dimension, $G(x_{1:t}) = \hat{x}_{t+1:t+T}$ is the concatenation of all predicted frames, and $D(.)$ is the discriminator network in adversarial loss. The discriminative loss in adversarial training is defined by

$$\begin{aligned} L_{Disc} = &-log D([x_{1:t}, x_{t+1:t+T}]) \\ &- log(1 - D([x_{1:t}, G(x_{1:t})])), \end{aligned} \tag{19}$$

where $x_{t+1:t+T}$ is the concatenation of all ground truth future frames. The generator G, corresponding to our proposed model for video prediction, aims to generate future frames that can mislead the discriminator D. The discriminator D, made up of four simple convolution layers, is trained to distinguish the predicted future frames from the truth frames as possible as it can. The training processes of generator and discriminator are executed alternately. That is to say, when the generator is trained, the discriminator is fixed and vice versa.

IV. Experimental Results

We evaluate the proposed approach on three diverse datasets: KTH [52], Human3.6M [53], and PennAction [54]. The public metrics mean Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index Measure (SSIM) are used for quantitative evaluation [6], [10]. Higher values of PSNR and SSIM indicate better performance. Moreover, the Learned Perceptual Image Patch Similarity (LPIPS) [55] is introduced to measure the perceptual distance between the predicted frames and ground truth. Lower LPIPS score means better similarity. Ablation studies are conducted to verify the effectiveness of each component. For all our experiments, we set $\alpha = 1$, $\beta = 0.02$ and $\gamma = 0.2$ in Eq. (14) and $\varsigma = 1$ in Eq. (10).

Network architecture. The architecture of our proposed model is illustrated in Fig. 2. Our motion encoder based on CSA is built similar to VGG16 [56] up to the third pooling layer, except that we replace its consecutive 3×3 convolutions with single 5×5, 5×5, and 7×7 convolutions in each layer as well as add an additional channel-wise and spatial attention module before the first pooling layer as described in section III-A.1. The content encoder is built with the same architecture as VGG16 up to the third pooling layer. The combine block consists of three consecutive 3×3 convolutions. The content decoder is the mirrored architecture of the content encoder where we perform unpooling followed by deconvolution, and a final tanh activation. The refinement with a soft attention map is like a gating mechanism. The soft attention block has the same architecture with the content decoder, except that the final activation is replaced by the sigmoid function. The proposed motion perceptual loss MPL as well as other loss are only used during training.

Baseline. We compare our method against seven methods, E3D-LSTM [27], ContextVP [12], EPVA [14], fRNN [13], MCnet [10], CDNA [6] and ConvLSTM [26]. We use the default implementation of fRNN [13] with some adaptations: The input of the model is resized to 128×128 since the input frame resolution we used is 128×128, and the numbers of nodes in each layer are changed reasonably. We use the default settings of MCnet [10] and re-train it by us. In order to fairly compare with other methods, we re-train CDNA [6] without scheduled sampling. The ConvLSTM [26] is widely used in many other methods and is trained with the same adversarial loss as our method.

A. KTH Dataset

The KTH human action dataset [52] consists of 600 videos of 15-20 seconds with 25 subjects performing 6 actions in

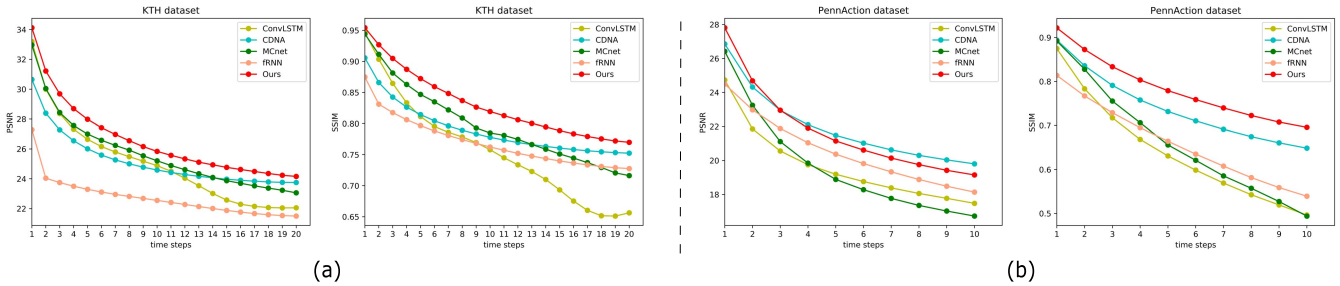


Fig. 5. Quantitative comparison between our method and other methods. (a) Evaluation on KTH dataset [52]. (b) Evaluation on PennAction dataset [54].

4 different settings. The actions contain 6 categories on a simple background: running, jogging, walking, boxing, hand-clapping and hand-waving. The videos are grayscale, at a resolution of 120 $\times$ 160 pixels and a frame rate of 25 fps. Following the paper [10], we use person 1-16 for training and person 17-25 for testing, and resize frames to 128 $\times$128 pixels. In the training, the network observes 10 frames and predicts the next 10 frames. In the testing, the network observes 10 frames and predicts 20 time steps into the future in order to verify the generalization ability of the model.

Quantitative comparisons. Fig. 5 (a) shows the quantitative results of our proposed method compared with other approaches. We can observe that (1) Our proposed method outperforms other four models in PSNR and SSIM by a large margin, thanks to more attention on dynamic areas in the low-level feature space and the supervision of high-level motion perceptual features. When predicting the frames at 20-th time step, our method produces video frames with the best PSNR/SSIM of 24.16/0.7699, compared to CDNA [6] (23.75/0.7524) and MCnet [10] (23.06/0.7163). (2) The performance of the MCnet [10] degrades more greatly in the long-term prediction, since it cannot capture the temporal motion cues. (3) The CDNA [6] seems to get better performance than MCnet [10] when the predicted time becomes long, but the qualitative results are very poor as shown in Fig. 6.

In addition, we also compute the average value of PSNR/SSIM over the 20 predicted frames, i.e., 26.61/0.831, which is lower than 29.3/0.879 of E3D-LSTM [27]. That may be due to the fact that E3D-LSTM recalls the long-range historical context by attending to previous distant memory states, which is very complicated and hard to train.

Qualitative results. Fig. 6 presents qualitative results of multi-step prediction on KTH dataset [52]. From the top to bottom we sequentially show the results of ConvLSTM [26], CDNA [6], MCnet [10], fRNN [13], our proposed model and ground truth. We can observe that (1) The prediction results of ConvLSTM [26] seem to have good performance in predicting the approximate motion trend, but loss the basic shape of a person. (2) When there are apparent movements in a video sequence, CDNA [6] and fRNN [13] cannot capture the dynamic information. Instead, they copy the last observed frame to the next future and produce some blurs in the moving space. (3) Compared with MCnet [10], our proposed method can preserve human detailed structures more accurately as shown in the green box. That is because our proposed channel-wise and spatial attention module can adaptively focus more

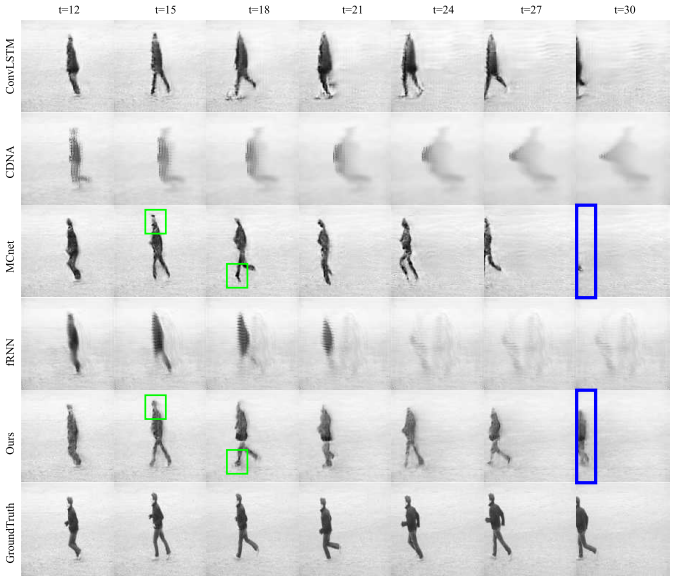


Fig. 6. Qualitative comparison on the KTH test set [52]. The results from top to down are generated by ConvLSTM [26], CDNA [6], MCnet [10], fRNN [13] and Ours. The last row is the ground truth. We show predictions from the 12^{th} frame, in every 3 time steps.

on key information of moving objects. (4) Moreover, our proposed method can capture the temporal motion cues and predict the accurate locations of moving objects in the long-term prediction compared with MCnet [10] as shown in the blue box. That results from the fact that our proposed motion perceptual loss based on high-level features can enforce the model to learn the motion trend perfectly.

B. Human3.6M Dataset

The Human3.6M dataset [53] consists of human actors performing various actions in a room. For training, we use subjects number 1, 5, 6, 7, 8, and test on subjects number 11. We subsample the video down to 10 fps such that there is noticeable motion in the video within reasonable time steps. Frame size is reduced to 128 $\times$ 128 or 64 $\times$ 64 by first removing 120 pixels from the left border and 100 pixels from the right border. In the training, we feed in 10 video frames and train all models to predict the next 10 time steps corresponding to 1 second. Our evaluations, including PSNR, SSIM as well as perceptual similarity metric LPIPS, measure performance up to 10 or 30 time steps into the future for the complex dataset.

TABLE I

EVALUATION (PSNR/SSIM/LPIPS) OF MULTI-FRAME (WITH 128 × 128 RESOLUTION) PREDICTIONS ON THE HUMAN3.6M DATASET [53]. THE BEST RESULTS ARE MARKED IN BOLD

Methods	Metric	T=2	T=4	T=6	T=8	T=10
ConvLSTM [26]	PSNR	28.23	25.73	24.59	23.59	22.58
	SSIM	0.9529	0.9301	0.9163	0.8995	0.8751
	LPIPS	0.0219	0.0333	0.0401	0.0473	0.0588
CDNA [6]	PSNR	29.23	26.82	25.55	24.73	24.13
	SSIM	0.9575	0.9370	0.9217	0.9099	0.9004
	LPIPS	0.0267	0.0431	0.0567	0.0663	0.0744
MCnet [10]	PSNR	30.05	26.55	24.94	23.90	22.83
	SSIM	0.9569	0.9355	0.9197	0.9030	0.8731
	LPIPS	0.0177	0.0284	0.0367	0.0462	0.0717
fRNN [13]	PSNR	27.58	26.10	25.06	24.26	23.66
	SSIM	0.9000	0.8885	0.8799	0.8729	0.8675
	LPIPS	0.0515	0.0530	0.0540	0.0539	0.0542
Ours	PSNR	**31.36**	**28.38**	**26.61**	**25.47**	**24.61**
	SSIM	**0.9663**	**0.9528**	**0.9414**	**0.9326**	**0.9235**
	LPIPS	**0.0151**	**0.0219**	**0.0287**	**0.0339**	**0.0419**

TABLE II

EVALUATION (PSNR/SSIM/LPIPS) OF LONG-TERM MULTI-FRAME (WITH 64 × 64 RESOLUTION) PREDICTIONS ON THE HUMAN3.6M DATASET [53]. THE BEST RESULTS ARE MARKED IN BOLD

Methods	Metric	T=1	T=6	T=12	T=18	T=24	T=30
ContextVP [12]	PSNR	**45.20**	-	-	-	-	-
	SSIM	**0.9960**	-	-	-	-	-
	LPIPS	-	-	-	-	-	-
EPVA [14]	PSNR	27.43	25.86	23.66	22.85	22.63	22.41
	SSIM	0.9509	0.9355	0.9150	0.9058	0.9027	0.8999
	LPIPS	0.0217	0.0267	0.0357	0.0404	0.0429	0.0460
EPVA-GAN [14]	PSNR	24.34	23.26	22.06	21.60	21.36	21.22
	SSIM	0.9300	0.9145	0.8986	0.8910	0.8869	0.8845
	LPIPS	0.0271	0.0320	0.0412	0.0461	0.0488	0.0512
Ours	PSNR	40.11	**29.94**	**26.49**	**25.98**	**24.87**	**23.73**
	SSIM	0.9907	**0.9616**	**0.9387**	**0.9345**	**0.9296**	**0.9265**
	LPIPS	**0.0035**	**0.0186**	**0.0337**	**0.0384**	**0.0410**	**0.0455**

Quantitative comparisons. TABLE I shows the quantitative comparisons (under the condition of 128 × 128 resolution) about our method, fRNN [13], MCnet [10], CDNA [6] and ConvLSTM [26] on Human3.6M dataset [53]. We can observe that (1) The CDNA has comparatively good predicted results, but it degrades greatly in the long-term prediction with PSNR/SSIM/LPIPS of 24.13/0.9004/0.0744 at 10-th time step. (2) Our proposed method gets the best performance in terms of PSNR, SSIM and LPIPS, especially in the case of relatively long-term prediction. At 10-th time step, the PSNR/SSIM/LPIPS of predictions is 24.61/0.9235/0.0419, increasing by 1.78/0.0504/0.0298 compared to MCnet [10]. The reasons for this are summarized as follows. Firstly, the Human3.6M dataset [53] is comparatively complex, with diverse backgrounds and motions. However, the MCnet [10] treats the motion features equally, which greatly degrades their discriminality. Secondly, our proposed channel-wise and spatial attention module can selectively emphasize the informative features in complicate scene.

Furthermore, we do some experiments on the condition that the videos are subsampled to 10 fps and downsampled to 64 × 64 resolution. TABLE II shows the comparison results of multi-frame predictions with ContextVP [12], EPVA [14] and EPVA-GAN [14]. The ContextVP [12] is limited to the short-term prediction and only do the next-frame prediction on Human3.6M dataset. We can observe that our method is inferior to ContextVP in the simple first frame prediction, but our method outperforms EPVA and EPVA-GAN in terms of PSNR and SSIM in all time steps, which means our model can not only get higher quantitative results but also generate realistic frames compared with the ground truth. Specially, compared to EPVA, our method achieves 1.32/0.0266 PSNR/SSIM improvement at 30-time step. In particular, our proposed model can not only get great performance with respect to PSNR/SSIM but also generate realistic future frames concerning LPIPS that have high perceptual similarity with ground truth. This is mainly due to the usage of motion perceptual loss that can enforce the model to learn more accurate high-level motion embedding and prompt to achieve relatively long-term video prediction.

Qualitative results. Fig. 7 shows qualitative examples of multi-step prediction on Human3.6M dataset [53]. For showing the problem clearly, we give the first five predicted results. It can be observed that (1) The prediction results of ConvLSTM and CDNA suffer from serious motion blur, and the static background cannot be restored perfectly because of the diverse scenes. (2) Compared to MCnet [10], our proposed method can preserve the static area perfectly as seen in the blue box. Compared to fRNN [13], our proposed method can not only retain the detailed structures of static area (as seen in the green box) but also restore the detailed structure of motion parts (as seen in the red box). Our proposed method works well even on the comparatively complex dataset, since it can selectively emphasize the key information of moving area and accurately distinguish the static and dynamic area to avoid the loss of prediction. (3) It demonstrates that hallucinating the prediction from scratch is difficult and copying pixels from nearby existing frames according to attention map is reasonable.

C. PennAction Dataset

The PennAction dataset [54] contains 2326 video sequences of 15 diverse actions as well as 13 human joint annotations for each sequence. Because of the motion complexity, the future is highly unpredictable. In our experiments, we use videos from the action of baseball pitch with 167 video sequences of 480 × 360. To train our network, we use the standard train split provided in the dataset and resize frames to 128 × 128 pixels. Our network is trained to observe 10 inputs and predict 10 time steps, and tested on predicting 10 time steps.

Quantitative comparisons. Fig. 5 (b) demonstrates the quantitative comparisons about our proposed method and other methods. We can observe that (1) The MCnet [10] is poor in the long-term video prediction. Since it adopts the simple frame difference to learn motion encoding, it is not robust to PennAction dataset with complex background, i.e., multiple motion objects or diverse scenes. (2) Our proposed method clearly outperforms all other methods in terms of SSIM by a large margin and is 0.0425 higher than CDNA [6] and 0.1971 higher than MCnet [10] at 10-th time step. This shows that our model is able to predict a structure similar to the ground truth image within the areas of motion. The PSNR results show

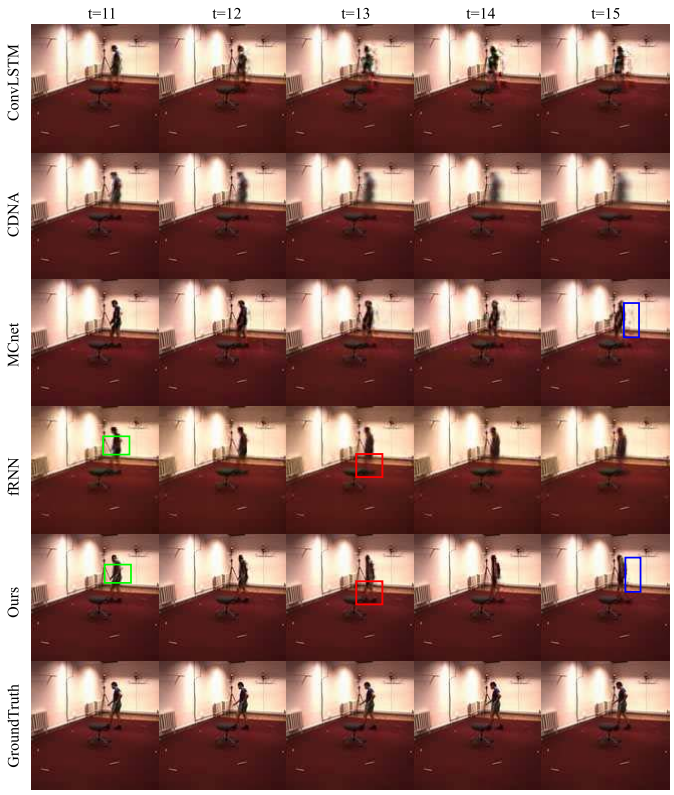


Fig. 7. Qualitative comparison on the Human3.6M test set [53]. The results from top to down are generated by ConvLSTM [26], CDNA [6], MCnet [10], fRNN [13] and Ours. The last row is the ground truth. All models are tested for predicting 10 time steps given 10 input frames. For better observation, we only show the first five predictions. Best view by zooming in.

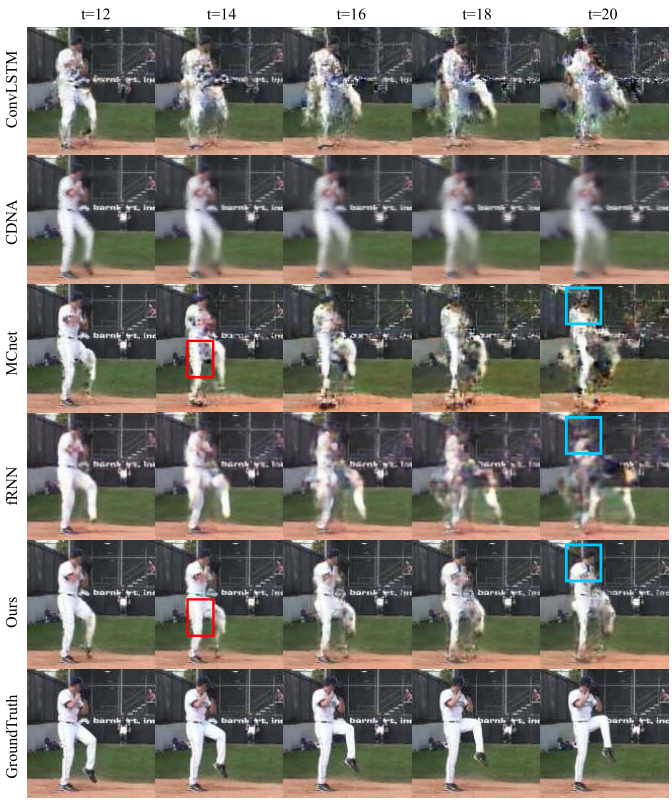


Fig. 8. Qualitative comparison on the PennAction test set [54]. The results from top to down are generated by ConvLSTM [26], CDNA [6], MCnet [10], fRNN [13] and Ours. The last row is the ground truth. All models are tested for predicting 10 time steps given 10 input frames.

that our method outperforms the other methods except for the CDNA [6]. Our method obtains the PSNR of 19.10 and is 0.7 lower than CDNA [6] at 10-th time step. The quantitative results are not enough to evaluate the performance of the video prediction. We further give some qualitative results in Fig. 8.

Qualitative results. From Fig. 8, we observe that (1) The CDNA [6] produces a future frame that looks like a person but cannot capture the dynamics, whose prediction is nearly stuck at all time steps. That is consistent with the above quantitative analyses that it is higher in terms of PSNR but lower in SSIM. (2) The MCnet [10] and fRNN [13] cannot capture the dynamic motion cues and produce serious motion blurs in the long-term prediction because of the complex motion patterns and diverse backgrounds. (3) Differently, our proposed method can retain the key detailed structure of motion area, i.e., areas in the red box, since our proposed channel-wise and spatial attention module can emphasize the informative features related to the motion structure in the complicated scenes. (4) Compared with MCnet [10] and fRNN [13], our proposed method can preserve the static area perfectly in the relatively long-term prediction as shown in the blue box. This strongly indicates that our method can accurately discriminate the relatively static areas even in the complex situations.

D. Ablation Study

Effectiveness of Channel-wise and Spatial Attention module. To analyse the effectiveness of the **C**hannel-wise and **S**patial **A**ttention (**CSA**) module for video prediction, we do some ablation studies on three datasets as shown in TABLE III, IV and V. To be specific, the model w/ CSA outperforms the baseline in terms of PSNR and SSIM on all datasets, especially on the complicate Human3.6M [53] and PennAction [54] datasets. At 10-th time step, the w/ CSA model obtains PSNR/SSIM of 23.73/0.9174 and 17.63/0.5722, with an increase of 0.9/0.0443 and 0.9/0.0783 compared with the baseline on Human3.6M [53] and PennAction [54] datasets respectively. That is due to the fact that our proposed CSA module emphasizes the key motion features in the low-level feature space, which can produce realistic predictions by working with attention map. Furthermore, when equipped with CSA, the gains got from the Motion Perceptual Loss are amplified on the complicated PennAction dataset at all time steps. Specially, the model w/ MPL_1 and w/ CSA obtains a salient increase of 2.08/0.1499 in terms of PSNR/SSIM at 10-th time step on PennAction dataset as shown in TABLE V compared with the model only w/ MPL_1 but w/o CSA.

Visualization of soft Attention Map. To further display the capability of our proposed channel-wise and spatial attention module in emphasizing the key information of moving objects, we present the attention map decoded from the motion high-level features on KTH dataset [52] as shown in Fig. 9. The first row shows the sequence of ground truth and the second row represents the corresponding soft attention map whose values belong to [0, 1]. The highlighted regions in the attention map

TABLE III

ABLATION STUDY (PSNR/SSIM) ON KTH [52] DATASET. **w/ MPL$_1$** DENOTES THE MODEL WITH ℓ_1 LOSS OF MOTION PERCEPTUAL FEATURES. **w/ MPL$_2$** MEANS THE MODEL WITH THE ℓ_2 LOSS OF MOTION PERCEPTUAL FEATURES. **w/ CSA** IS THE MODEL WITH THE CHANNEL-WISE AND SPATIAL ATTENTION MODULE AS WELL AS THE ATTENTION MAP

Model		T=4	T=12	T=20
w/o CSA	w/o MPL	27.57 / 0.8628	24.61 / 0.7746	23.06 / 0.7163
	w/ MPL$_1$	28.28 / 0.8728	24.99 / 0.7857	23.76 / 0.7422
	w/ MPL$_2$	28.19 / 0.8753	24.97 / 0.7784	23.98 / 0.7212
w/ CSA	w/o MPL	28.37 / 0.8783	24.81 / 0.7817	23.47 / 0.7311
	w/ MPL$_1$	28.70 / 0.8873	25.34 / 0.8061	24.16 / 0.7699

TABLE IV

ABLATION STUDY (PSNR/SSIM) ON HUMAN3.6M [53] DATASET

Model		T=2	T=6	T=10
w/o CSA	w/o MPL	30.05 / 0.9569	24.94 / 0.9197	22.83 / 0.8731
	w/ MPL$_1$	30.14 / 0.9559	25.87 / 0.9302	23.77 / 0.9025
	w/ MPL$_2$	30.19 / 0.9582	25.66 / 0.9303	23.22 / 0.8919
w/ CSA	w/o MPL	31.29 / 0.9670	26.06 / 0.9376	23.73 / 0.9174
	w/ MPL$_1$	31.36 / 0.9663	26.61 / 0.9414	24.61 / 0.9235

TABLE V

ABLATION STUDY (PSNR/SSIM) ON PENNACTION [54] DATASET

Model		T=2	T=6	T=10
w/o CSA	w/o MPL	23.24 / 0.8277	18.28 / 0.6209	16.73 / 0.4939
	w/ MPL$_1$	22.59 / 0.8253	18.35 / 0.6428	17.02 / 0.5411
	w/ MPL$_2$	22.34 / 0.8155	18.30 / 0.6289	16.95 / 0.5199
w/ CSA	w/o MPL	24.35 / 0.8561	19.48 / 0.6827	17.63 / 0.5722
	w/ MPL$_1$	24.60 / 0.8711	20.56 / 0.7549	19.10 / 0.6910

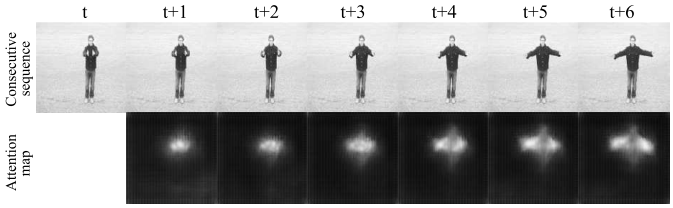


Fig. 9. The soft attention map based on enhanced motion-aware features on KTH dataset [52].

correspond to the dynamic areas that should be paid more attention. The visualization of a soft attention map shows its discriminability. On the one hand, it benefits from the proposed CSA for selecting informative features related to motion. On the other hand, it is reasonable to explicitly copy from the previous frame for non-highlighted regions.

Effectiveness of Motion Perceptual Loss. The ablation studies of our proposed **M**otion **P**erceptual **L**oss (**MPL**) are shown in TABLE III, IV and V. Here **w/ MPL$_1$** denotes the model with ℓ_1 loss of motion perceptual features. **w/ MPL$_2$** means the model with the ℓ_2 loss of motion perceptual features. We can observe that the model with MPL$_1$ (MPL$_2$) has a significant improvement on KTH [52] and Human3.6M [53] datasets. Especially, the model w/ MPL$_1$ gets 23.76/0.7422 PSNR/SSIM at 20-th time step, with an increase of 0.6/0.0259 compared to the baseline on KTH dataset. It achieves 23.77/0.9025 PSNR/SSIM at 10-th time step, 0.94/0.0294 higher than the baseline on the Human3.6M dataset. For PennAction dataset [54] as shown in TABLE V, the models with MPL$_1$ or MPL$_2$ only get better performance after the 4-th time step when not equipped with CSA. However, when equipped with CSA, the model with MPL$_1$ obtains significant improvements over the model without MPL at all time steps, especially with an increase of 1.47/0.1188 PSNR/SSIM at 10-th time step, which greatly benefits from the discriminative motion features obtained from CSA. That strongly illustrates the proposed motion perceptual loss can make full use of the high-level motion feature as a signal to enforce the model to learn more the motion trend for robust long-term prediction.

TABLE VI

QUANTITATIVE RESULTS (PSNR/SSIM) FOR PREDICTING THE PAST ON KTH DATASET [52]. THE BEST RESULTS ARE MARKED IN BOLD

Methods	T=30	T=21	T=12
MCnet [10]	**30.75** / **0.9173**	**25.05** / 0.7872	23.41 / 0.7296
Ours	30.16 / 0.9171	24.98 / **0.7901**	**23.74** / **0.7495**

Comparison of different loss functions. Most existing approaches [10], [14], [15] of video prediction fit ℓ_2 loss in their models. The ℓ_2 loss is known to penalize large pixel errors while oversmooth the structural details [57]. We further explore different loss functions, i.e., MPL$_2$ based on ℓ_2 loss and MPL$_1$ based on ℓ_1 loss. The results shown in TABLE III, IV and V demonstrate the model with MPL$_1$ is superior to ones with MPL$_2$ in most time steps. The reason for the relatively poor performance of MPL$_2$ is that the ℓ_2 loss extremely penalizes large errors, which obviously leads to the unbalanced concerns about different features.

E. Predicting the Past

To verify the generalization ability of our proposed model, we conduct a novel comparison experiment, referred as predicting the past similar to [11]. The model is trained with temporal sequential order, but is tested with temporal reverse order. It is difficult because reverse motion patterns are never learnt by the model. TABLE VI shows the quantitative results for predicting the past on KTH dataset [52]. The best results are marked in bold. Our proposed method has a better performance than the MCnet [10] in the long-term video prediction in terms of PSNR and SSIM. Fig. 10 shows the prediction with temporal reverse order. To be specific, (1) The MCnet [10] fails to give reasonable predictions on the reverse order, i.e., the arms are totally warped compared with the ground truth. (2) Our proposed method produces realistic looking and sensible motion even on the reverse order, which clearly shows the ability of the channel-wise and spatial attention module in emphasizing the key information and the motion perceptual loss in learning the essential temporal motion cues.

F. Complexity Analysis

To evaluate the complexity of the model, we count the number of our proposed model parameters, i.e., 14.1M and the number of E3D-LSTM parameters, i.e., 28.5M. Although the E3D-LSTM [27] gets better performance on the simple

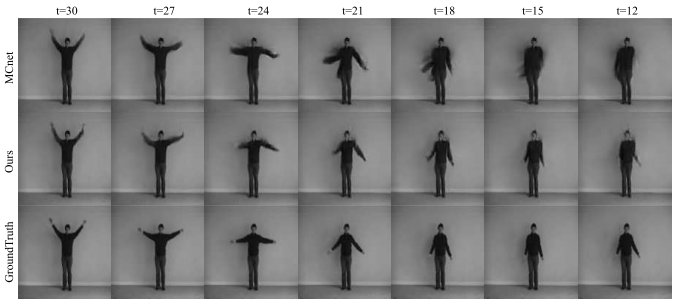


Fig. 10. Qualitative comparison for predicting the past on the KTH dataset [52]. All models are tested for predicting 20 time steps given 10 input frames. Best view by zooming in.

cyclical KTH dataset, its number of parameters is more than twice ours, which needs more resources and is very hard to train. Our proposed model tends to be stable after the 30000-th iteration. After obtaining the trained model, it can predict one future frame with about 0.21 second on the test set.

V. Conclusion

In this paper, we propose a novel video prediction method named Motion-Aware Feature Enhancement network. The proposed channel-wise and spatial attention module can emphasize the informative motion-aware features in the low-level space, which can greatly enhance the discriminative capability of soft attention map. Moreover, the proposed motion perceptual loss uses the high-level motion features as supervision to enforce the model to learn more the relatively long-term motion trends. Extensive experimental results on three human activity video datasets: KTH [52], Human3.6M [53], and PennAction [54] demonstrate that our proposed approach outperforms state-of-the-art methods under different evaluation metrics.

References

[1] Z. Liu, R. A. Yeh, X. Tang, Y. Liu, and A. Agarwala, "Video frame synthesis using deep voxel flow," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4473–4481.

[2] W. Lotter, G. Kreiman, and D. D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–18.

[3] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 843–852.

[4] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 14–29, Jan. 2016.

[5] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 201–214.

[6] C. Finn, I. J. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2016, pp. 64–72.

[7] X. Jin *et al.*, "Predicting scene parsing and motion dynamics in the future," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6918–6927.

[8] F. Ebert, C. Finn, A. X. Lee, and S. Levine, "Self-supervised visual planning with temporal skip connections," in *Proc. Conf. Robot Learn.*, 2017, pp. 344–356.

[9] J. Oh, X. Guo, H. Lee, R. L. Lewis, and S. P. Singh, "Action-conditional video prediction using deep networks in atari games," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2015, pp. 2863–2871.

[10] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–22.

[11] J. Xu, B. Ni, Z. Li, S. Cheng, and X. Yang, "Structure preserving video prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1460–1469.

[12] W. Byeon, Q. Wang, R. K. Srivastava, and P. Koumoutsakos, "Contextvp: Fully context-aware video prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 781–797.

[13] M. Oliu, J. Selva, and S. Escalera, "Folded recurrent neural networks for future video prediction," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 745–761.

[14] N. Wichers, R. Villegas, D. Erhan, and H. Lee, "Hierarchical long-term video prediction without supervision," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 6033–6041.

[15] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee, "Learning to generate long-term future via hierarchical prediction," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3560–3569.

[16] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1933–1941.

[17] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal residual networks for video action recognition," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3468–3476.

[18] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "MARS: Motion-augmented RGB stream for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7882–7891.

[19] Z. Shou *et al.*, "DMC-net: Generating discriminative motion cues for fast compressed video action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1268–1277.

[20] J. Y.-H. Ng, J. Choi, J. Neumann, and L. S. Davis, "ActionFlowNet: Learning motion representation for action recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1616–1624.

[21] J. Johnson, A. Alahi, and L. Feifei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.

[22] T. Wang *et al.*, "Video-to-video synthesis," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1144–1156.

[23] J. Pan *et al.*, "Video generation from single semantic label map," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3733–3742.

[24] Y. Jang, G. Kim, and Y. Song, "Video prediction with appearance and motion conditions," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1–14.

[25] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[26] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[27] Y. Wang, L. Jiang, M. Yang, L. Li, M. Long, and L. Feifei, "Eidetic 3D LSTM: A model for video prediction and beyond," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–14.

[28] J. Xu, B. Ni, and X. Yang, "Video prediction via selective sampling," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1705–1715.

[29] R. F. Emily Denton, "Stochastic video generation with a learned prior," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1–12.

[30] M. Babaeizadeh, C. Finn, D. Erhan, R. H. Campbell, and S. Levine, "Stochastic variational video prediction," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–15.

[31] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine, "Stochastic adversarial video prediction," 2018, *arXiv:1804.01523*. [Online]. Available: http://arxiv.org/abs/1804.01523

[32] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[33] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2016, pp. 613–621.

[34] J. Yosinski, J. Clune, A. Nguyen, T. J. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," 2015, *arXiv:1506.06579*. [Online]. Available: https://arxiv.org/abs/1506.06579

[35] C. Szegedy *et al.*, "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–10.

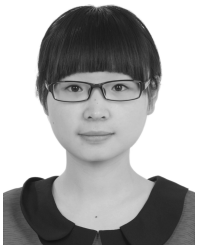
[36] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 427–436.
[37] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5188–5196.
[38] P. Weinzaepfel, H. Jegou, and P. Perez, "Reconstructing an image from its local descriptors," in *Proc. CVPR*, Jun. 2011, pp. 935–938.
[39] C. Vondrick, A. Khosla, H. Pirsiavash, T. Malisiewicz, and A. Torralba, "Visualizing object detection features," *Int. J. Comput. Vis.*, vol. 119, no. 2, pp. 145–158, Sep. 2016.
[40] L. A. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2015, pp. 262–270.
[41] L. Gatys, A. Ecker, and M. Bethge, "A neural algorithm of artistic style," *J. Vis.*, vol. 16, no. 12, p. 326, Sep. 2016.
[42] Z. Yang, Y. Li, J. Yang, and J. Luo, "Action recognition with spatio–temporal visual attention on skeleton image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2405–2415, Aug. 2019.
[43] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Generating images from captions with attention," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–4.
[44] L. Chen *et al.*, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5659–5667.
[45] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 2048–2057.
[46] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Nov. 2018, pp. 7132–7141.
[47] F. Wang *et al.*, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3156–3164.
[48] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.
[49] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3085–3094.
[50] Y. Hu, J. Li, Y. Huang, and X. Gao, "Channel-wise and spatial feature modulation network for single image super-resolution," *IEEE Trans. Circuits Syst. Video Technol.*, early access, May 7, 2019, doi: 10.1109/TCSVT.2019.2915238.
[51] K. Su, D. Yu, Z. Xu, X. Geng, and C. Wang, "Multi-person pose estimation with enhanced channel-wise and spatial information," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5674–5682.
[52] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, 2004, pp. 32–36.
[53] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1325–1339, Jul. 2014.
[54] W. Zhang, M. Zhu, and K. G. Derpanis, "From actemes to action: A strongly-supervised representation for detailed action understanding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2248–2255.
[55] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.
[56] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[57] B. Huang and H. Ling, "End-to-end projector photometric compensation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6810–6819.

Xue Lin received the B.S. and M.E. degrees from the School of Information Science and Engineering, University of Jinan, Jinan, China, in 2015 and 2018, respectively. She is currently pursuing the Ph.D. degree with the Department of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. Her research interests are computer vision, image processing, and machine learning, as well as their applications on human-centric activity understanding.

Qi Zou received the Ph.D. degree in computer science from Beijing Jiaotong University, Beijing, China, in 2006. She is currently a Professor and a Ph.D. Supervisor with the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests include computer vision and intelligent transportation systems.

Xixia Xu received the B.S. degree in software engineering from Lanzhou Jiaotong University, Lanzhou, China, in 2018. She is currently pursuing the Ph.D. degree with the Department of Computer and Information Technology, Beijing Jiaotong University, Beijing, China, in 2019. Her current research interests include computer vision, image processing, and machine learning with the applications on 2D multi-person pose estimation analysis and human-centric behavior analysis.

Yaping Huang received the B.S., M.S., and Ph.D. degrees from the School of Computer and Information Technology, Beijing Jiaotong University, China, in 1995, 1998, and 2004, respectively. She is currently a Professor with the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests are computer vision, machine learning, and pattern recognition.



Yi Tian received the B.E. and Ph.D. degrees from Beijing Jiaotong University, China, in 2011 and 2018, respectively. She was a Visiting Student with the Electrical and Computer Engineering Department, Northeastern University, Boston, MA, USA, from 2016 to 2017. She is currently a Lecturer with the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests mainly focus on zero-shot learning and human action recognition.