Multi-view Action Recognition using Cross-view Video Prediction

Shruti Vyas, Yogesh S Rawat, and Mubarak Shah

Center for Research in Computer Vision, University of Central Florida, USA
{shruti,yogesh,shah}@crcv.ucf.edu

Abstract. In this work, we address the problem of action recognition in a multi-view environment. Most of the existing approaches utilize pose information for multi-view action recognition. We focus on RGB modality instead and propose an unsupervised representation learning framework, which encodes the scene dynamics in videos captured from multiple viewpoints via predicting actions from unseen views. The framework takes multiple short video clips from different viewpoints and time as input and learns an holistic internal representation which is used to predict a video clip from an unseen viewpoint and time. The ability of the proposed network to render unseen video frames enables it to learn a meaningful and robust representation of the scene dynamics. We evaluate the effectiveness of the learned representation for multi-view video action recognition in a supervised approach. We observe a significant improvement in the performance with RGB modality on NTU-RGB+D dataset, which is the largest dataset for multi-view action recognition. The proposed framework also achieves state-of-the-art results with depth modality, which validates the generalization capability of the approach to other data modalities. The code is publicly available at https://github.com/svyas23/cross-view-action.

1 Introduction

Historically, viewpoint in-variance has been a very active research area in computer vision and is currently also important from the perspective of representation learning. The appearance and dynamics of action vary from one viewpoint to another. Humans have the ability to effortlessly visualize how action might look like from an unseen viewpoint. This ability highlights the view-invariant property of the encoded representation in the human brain after observing the action from certain views [13]. The encoded representation should have sufficient details to predict the dynamics of the action from unseen views. Motivated by this, we present an unsupervised representation learning framework which encodes the scene dynamics in videos captured from multiple viewpoints via predicting actions from unseen views. An overview of the proposed framework is shown in Figure 1. The prediction of videos from unseen viewpoint and time enforces the network to learn a more informative view and time-dependent representation, which makes it effective for multi-view environment.

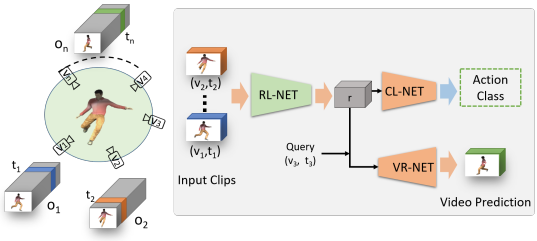

Fig. 1. An overview of the proposed representation learning framework. An action is captured from different viewpoints (v1, v2, v3, ..., vn) providing observations (o1, o2, o3, ..., on). Video clips from two viewpoints (v1 and v2) at arbitrary times (t1 and t2) are used to learn a representation (r) for this action, employing the proposed representation learning network (RL-NET). The learned representation (r) is then used to render a video from an arbitrary query viewpoint (v3) and time (t3) using proposed video rendering network (VR-NET). The representation thus learned is used for action recognition using classification network (CL-NET)

Cross-view prediction is a challenging problem as the visual appearance and dynamics of an action vary from one viewpoint to another. There have been some efforts in this direction for novel-view image prediction, which includes 3D reconstruction from images [14] and cross-view image rendering [36,8,48]. However, most of the existing research in video prediction mainly focus on single view videos. Video prediction in itself is a challenging problem [5], and the variation in actions observed from different viewpoints makes it more challenging. In this work, we explore the idea of cross-view video prediction to learn a good representation for action recognition in a multi-view environment.

There has been a lot work in view-invariant action recognition in recent years. However, most of the existing works make use of skeleton data to learn the view-invariant features [51]. Learning a view-invariant representation is much more challenging with RGB videos as compared to skeleton sequences, and therefore, the methods using RGB modality do not perform as well as the skeleton-based methods [27,52]. Researchers have also explored the use of other modalities such as depth and optical flow for view-invariant learning, but they are also not very effective and do not generalize well to RGB modality [22]. We propose a framework to learn a robust feature representation from RGB videos.

The proposed framework takes multiple short video clips from different viewpoints and times as input and learns an internal representation using the proposed representation learning network (RL-NET). The learned representation is used to predict a video clip from an unseen viewpoint and time with the help of a video rendering network (VR-NET) and also used for action recognition using a classification network (CL-NET). The proposed framework is adaptive for number of input views which makes it suitable for multi-view as well as novel-view action recognition. We demonstrate its effectiveness in both cross-subject as well as cross-view action recognition. Moreover, it is also effective in integrating mul-

tiple views to further improve the action recognition performance. We make the following contributions in this work:

- We propose a framework for *unsupervised multi-view* representation learning via *cross-view* video prediction which can be trained end-to-end from scratch.
- We propose a viewpoint and time conditioned encoding of videos which are integrated to get a *holistic representation* of the action from multiple views. This allows the network to preserve the notion of *viewpoint and time*, which facilitates query-based cross-view prediction.
- We evaluate the effectiveness of the proposed approach with *multiple modalities* and on multiple datasets. We observe a significant improvement in the performance of cross-view as well as cross-subject action recognition on the largest multi-view dataset when compared with existing methods.

2 Related Work

Cross-view Action Recognition: We have recently seen a good progress in action recognition with RGB videos [4,47]. However, these methods are mainly focused on single view videos. In multi-view environment, the availability of different modalities such as, pose and depth, has motivated many research works which address the issue of view in-variance. In this stream of research, most of the works make use of multiple modalities (RGB+D) [39], depth [34,22], RGB [27,22,52] or skeleton data [21,51,46] to learn view-invariant features. Among these, skeleton data has shown very promising performance when compared with RGB videos. The existing methods utilize RNN [7,25,24,2,51], CNN [24,15,26], and GraphCNN [49,21] to learn view-invariant features from skeleton sequences. However, these approaches requires the availability of 2D/3D pose information. Apart from these, 3D motion is another modality which has shown good results in cross-view action recognition [27,22]. However, getting 3D motion is computationally expensive and these methods do not generalize well with RGB modality. In this work, we focus on RGB modality for multi-view action recognition.

Cross-View prediction: The research in image prediction using deep learning has recently seen a great progress [18], and it is mainly attributed to the success of Generative Adversarial Networks (GAN). We have also seen some preliminary success in the research on video prediction and proposed methods are mainly focused on future frame prediction [30,3], future clip prediction [42], or conditioned video generation [41,37]. Our work is different from these approaches as we have a notion of viewpoint, which has not been addressed earlier. Cross-view prediction of data is an interesting problem, which can have multiple applications including view-invariant representation learning. There are some existing works focusing on this problem for cross-view image prediction [36,8] and 3D reconstruction from images [14]. In [8], the authors proposed a scene representation learning framework and worked with synthetic images. In a recent effort [17], the authors propose a novel view video prediction approach which requires a strong prior, depth/skeleton sequence, from the target view for video

generation. Our work is different from this in two main aspects; our approach predicts the target video based on a query viewpoint and does not require any prior from the target view, and we are using cross-view prediction as an auxiliary task for unsupervised learning of action representation in a multi-view environment.

Unsupervised Representation Learning: The research in unsupervised video representation learning mainly focuses on encoder-decoder type of networks, where the decoder is used for reconstruction [10] or predicting future frames [40]. There are some other approaches which utilize temporal ordering of clips [9,31], temporal coherence [45], and sorting of shuffled frames [19] as a way of unsupervised learning. Recently, there has been some effort to utilize 3D-motion prediction as a way of unsupervised learning [27,22], but this requires computation of optical flow which is computationally costly. The existing works are mostly focused on single views and they are not effective for multi-view learning. We are focusing on unsupervised learning in multi-view environment.

3 Method

The proposed framework consists of two main components, a representation learning network (RL-NET), f, and a video rendering network (VR-NET), g. A detailed overview of the proposed framework is shown in Figure 2. The input to the framework consists of multiple short video clips of an instance captured from varying viewpoints and time which are termed as observations, $o_i = \{(x_i^k, v_i^k, t_i)\}_{k=1,2,...,K}$, where, x_i^k represents k^{th} video clip captured from viewpoint v_i^k and time t_i for any instance i. The RL-NET take these observations as input and learns a holistic representation r for the instance with the help of an encoding network (ENC-NET), f^e, and a blending network (BL-NET), f^b, preserving the notion of view and time.

The ENC-NET considers each observation independently and encode the spatio-temporal features integrated with viewpoint and time, $e_i^k = f^e(o_i^k)$. Here e_i^k is the view and time dependent encoding for observation o_i^k from instance i. BL-NET is a recurrent network which updates its internal representation as it sees more observations before providing a holistic representation $r_i = f^b(\{e_i^k\}_{k=1,2,...,K})$ of the scene and its dynamics. The VR-NET, then, use this representation, r, along with stochastic latent variable, $z \sim (0, 1)$, to render a video clip from a query viewpoint, v_i^k, and time, t_i.

Formally, we can define the representation learning as, $r_i = f_\theta(o_i)$, and the video rendering network as,

$$g_\theta(x|v^q, t^q, r) = \int g_\theta(x, z|v^q, t^q, r)dz, \tag{1}$$

where, $g_\theta(x|v^q, t^q, r)$ is probability density of a video x observed from a query viewpoint v^q at time t^q, for an instance o_i with representation r and latent variable z. The parameters θ for the two networks can be learned using optimization over the rendered video. We train the two networks, RL-NET and

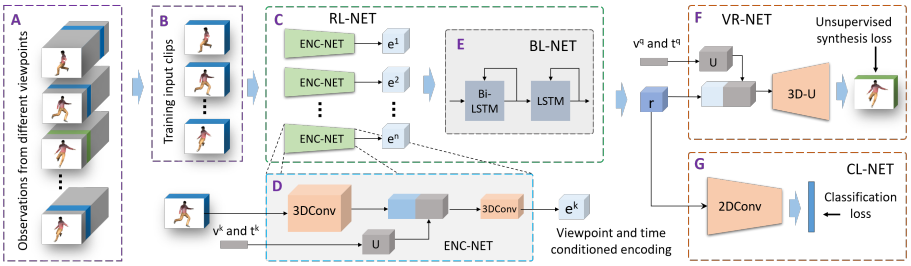

Fig. 2. Outline of the proposed unsupervised cross-view video rendering framework. **A** : A collection of observations (o) for a given action from different viewpoints. **B**: Training clips from the set of observations captured from different viewpoints and at different times. **C**: Representation learning network (RL-NET), which takes video clips from different viewpoint and time as input and learns a representation r. **D**: ENC-NET is used to learn individual video encodings e^k conditioned on its viewpoint v^k and time t^k. **E**: The blending network (BL-NET) combines encodings learned from different video clips into a unified representation r. **F**: The representation r is used to predict a video from query viewpoint v^q and time t^q using VR-NET. **G**: The representation r can also be used for action classification using CL-NET. 3D-U refers to 3D convolutions combined with upsampling and U refers to upsampling.

VR-NET, jointly in an end-to-end fashion to maximize the likelihood of rendering the ground-truth video, observed from the query viewpoint and time. In the next subsections we will describe each of these components in detail (More details are in the supplementary).

3.1 Representation Learning Network (RL-NET)

The representation learning network f takes multiple video clips of an instance captured from different viewpoints and time to learn a representation r. It consists of two components, encoding network (ENC-NET) f^e and a blending network (BL-NET) f^b.

Encoding Network (ENC-NET): The ENC-NET f^e learns spatio-temporal features for each observations independently using 3D convolutions. Since each observation comes from a different viewpoint and time, we want to integrate these factors in the learned encodings. The ENC-NET first extracts viewpoint and time independent features e^{hk} from the input video clip x^k; $e^{hk} = f^{eh}(x^k)$. Here f^{eh} is a 3D convolution network with two layers. This encoding is then passed to a integration network f^{ei} along with viewpoint v^k and time t^k encodings. The integration network gives us a viewpoint and time dependent encodings $e^k = f^{ei}(e^{hk}, v^k, t^k)$. It takes the viewpoint and time encodings and upsample them to match with the shape of the features extracted by f^{eh}. Then they are concatenated together along the channels axis before passing to a 3D convolution network which consists of five more layers. The ENC-NET can be represented as

$e^k = f^e(o^k) = f^{ei}(f^{eh}(x^k), v^k, t^k)$, and it is shared among all the observations of an instance during training.

Viewpoint and Time Integration: A viewpoint of an observation is defined using two different parameters: camera position and its orientation. The camera position is defined by its location which includes height h^v, distance d^v, and angular position a^v with respect to the actor. The height and distance values are normalized between $(0, 1)$. The angular position is encoded depending upon where the viewpoint is positioned and it lies in the range (-π, π). The orientation is defined by horizontal-pan hp^v and vertical-pan vp^v in the camera while capturing the observation. This is taken into account when we randomly crop each input observation in the spatial dimension during training. The time encoding t^e is derived using the position of the observation in the long action sequence. It is normalized in the range $(0, 1)$. The viewpoint and time encodings becomes a six dimensional vector, where $v = [h^v, d^v, a^v, hp^v, vp^v]$, and $t = [t^e]$, which is used to get a viewpoint and time integrated video features $e^k = f^e(x^k, v^k, t^k)$ for a given observation o^k.

Blending Network (BL-NET): The viewpoint and time conditioned encodings e^k from each observation are passed to a blending network f^b which learns a representation r. We want to learn a representation which holistically represents the scene and its dynamics as viewed from different viewpoints and time. The recurrent networks, such as LSTM, have been widely used to learn temporal dependencies in sequential data. However, they are also shown to be effective in processing non-sequential data, such as addition of a series of numbers [12]. Motivated by this, we propose a recurrent network which learns a representation for an instance, which is updated as it sees new observations (Figure 2E). More specifically, we utilize an LSTM architecture, where the memory cell, c, acts as an accumulator of state information and is updated by the input (i), output (o) and forget (f) gates, which are self-parameterized controlling gates. We make use of convolutional LSTM to preserve the spatial information in the embeddings and utilize bi-directional layers in the network for a more effective learning. For a given video embedding, e_i^k, after seeing all other observations in a forward and a backward pass, we get an updated hidden representation h_i^r.

$$h_i^r = (o_i^f \circ \tanh(c_i^f)) \frown (o_i^b \circ \tanh(c_i^b)). \tag{2}$$

Here, o_i^f and o_i^b are the output gates of the forward pass and backward pass respectively, c_i^f and c_i^b are the corresponding memory cell states, $\circ$ denotes the Hadamard product, and $\frown$ denotes a concatenation operation between learned representations from the forward and backward pass. The updated intermediate representation from each observation is then passed to a uni-directional conv-LSTM layer, which integrate these to get a holistic representation r.

$$r = o_n \circ \tanh(c_n). \tag{3}$$

Here, o_n is the output gate, c_n is the memory cell state of the network after seeing all the n observations. The learned representation can be computed as $r = f^b(\{f^e(o^k)\}_{k=1,2...n})$.

3.2 Video Rendering Network (VR-NET)

The representation, r, learned with the given observations, o, is used to render a video with a video rendering network (VR-NET). The VR-NET, shown in Figure 2F, is also a convolution based network which takes as input the learned representation, r, along with query viewpoint, v^q, time t^q, and latent noise z. The viewpoint v^q, time t^q, and noise z are passed to the network as conditioning, for which we use concatenation operation with the representation features. The idea is to extract viewpoint and time dependent features from the learned representation r. The VR-NET consists of 2D convolutions followed by 3D convolutions to render the video clips. The convolution layers are used in combination with upsampling of features to generate video clips with resolution same as the input observations. The video rendering is represented as $V^p = g(r, v^q, t^q, z) = g(f(o), v^q, t^q, z)$.

The two networks, RL-NET and VR-NET, are trained jointly in an end-to-end fashion minimizing the reconstruction loss L^r as the objective function. The reconstruction loss L^r is computed as mean squared error between the predicted video V^p and the ground truth video clip V^g.

$$L^r = \frac{1}{N}\sum_{n}^{N}\sum_{i}^{F}\sum_{j}^{H}\sum_{k}^{W}\sum_{m}^{C}||V^p_{ijkm} - V^g_{ijkm}||^2. \tag{4}$$

Here, N is the number of samples, F is the number of frames in the clip, H, W is height and width of the video frames, and $C = 3$ for three RGB color channels.

3.3 Action Recognition

The RL-NET and VR-NET can be trained jointly for unsupervised representation learning by cross-view prediction. To explore the effectiveness of the learned representation, we use it for the task of cross-view action recognition using a supervised approach. We use the same RL-NET and VR-NET framework and add a classifier (CL-NET) on top of learned representation. CL-NET has 2 convolution layers followed by fully connected layers and it predicts probabilities for each action classes. We use categorical cross entropy to compute the loss L^c for the action recognition.

$$L^c = -\frac{1}{N}\sum_{n}^{N}\sum_{c}^{C} 1_{y_i \in C_c} \log(\hat{p}[y_i \in C_c]). \tag{5}$$

Here, C is the number of action categories, and $\hat{p}[y_i \in C_c]$ is the predicted probability for this sample corresponding to category c.

The proposed framework can be trained in two different ways for action classification. In the first approach, we have a two step process where we first train RL-NET and VR-NET in an unsupervised way to learn a representation. In the next step, we train a CL-NET using this representation for action classification. In the second approach, we train all the three networks, RL-NET, VR-NET, and

CL-NET, in a joint training. In our preliminary experiments, we observe similar action recognition performance with both the approaches. In all our reported experiments, we follow the second approach as it is efficient in terms of time due to a joint single step training.

The network is trained end-to-end with the two loss functions (L^r and L^c) in a multi-task setting and the overall loss of the network is defined as,

$$L = \lambda_r \times L^r + \lambda_c \times L^c. \tag{6}$$

In all our experiments, we use $\lambda_r = \lambda_c = 1.0$. The network is trained using observations captured from certain known views and later tested on observations from unseen views for cross-view action recognition.

3.4 Training and Implementation Details

We train the proposed network without any pre-trained weights using Adam optimizer and a learning rate of $2e^{-5}$. A batch-size of 6 was used in all our experiments. The input video clips to RL-NET consists of 6 frames with a skip rate of 3 and a resolution of 112x112. The network takes 6 video clips at a time and renders one video clip with 6 frames and a resolution of 112x112 during training. We implemented our code in Keras with Tensorflow backend and use Titan-X GPU for training our network.

4 Experiments

We perform our experiments on two different datasets: NTU-RGB+D [38] and Northwestern-UCLA MultiviewAction3D (N-UCLA) [44]. We use NTU-RGB+D dataset for all our ablation studies.

NTU-RGB+D: This human action recognition dataset contains more than 56K videos and 4 millions frames with 60 different actions. There are a total of 40 different actors, who perform actions captured from 80 different viewpoints. We perform both cross-subject (CS) and cross-view (CV) evaluation for action recognition as suggested by [38] on RGB as well as depth modality. For cross-view video prediction experiments, we use the subject split suggested by [38].

Northwestern-UCLA MultiviewAction3D (N-UCLA): This dataset has 10 action categories and each action is performed by 10 actors. The actions are captured from 3 viewpoints and there are a total of 1493 action sequences. We perform both CS and CV evaluation as suggested by [44] and use videos from the first two views for training and videos from the third view for testing. This is a much smaller dataset in comparison with NTU-RGB+D and we use this in our transfer learning experiments for action recognition.

4.1 Representation Learning via Rendering

The proposed method utilize video prediction for learning a representation. We experimented with three different scenarios of rendering during training. These

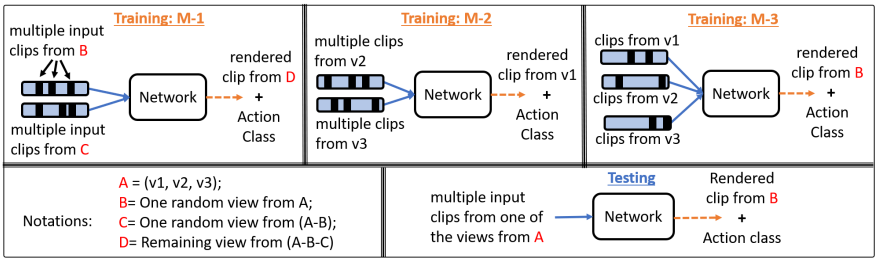

Fig. 3. Details of different training strategies (M-1, M-2, and M-3) which are used to study the effect of video rendering on representation learning for action classification. All the three variations use the same testing strategy

Table 1. A comparison of classification accuracy from different training configurations to study the effect of rendering on cross-subject split of NTU-RGB+D dataset. These evaluations are done on only 6 clips per video which is similar to the training setup

Training Approach	Accuracy			
	Testing View			Average
	view 1	view 2	view 3	
M-1	77.3	74.2	72.2	74.6
M-2	59.8	59.7	58.4	59.3
M-3	57.3	56.2	55.3	56.3

scenarios are based on what the network sees as input and what it tries to render. In the first scenario (M-1), the network sees all the available views as input except one, which is used for rendering. The input views are selected randomly, therefore, eventually the network will see all the available views as input. In the second scenario (M-2), one view is kept for rendering and the rest are used as input throughout the training. In the third scenario (M-3), all the views are used a input and one view and time is randomly selected for rendering. In this case, the network will render a seen view however it may be from a different time.

In NTU-RGB+D dataset, there are three different viewpoints. Therefore, for the first variation (M-1), we randomly select two input views and render a video from third unseen view. In the second configuration (M-2), we fixed the input views to 2 and 3, and view 1 is used for rendering. And, in the last configuration (M-3), we use all the three views as input and randomly select one of them for rendering at a random time-step. The details of these configurations are shown in Figure 3.

We analyze the performance of these configurations for action recognition. We use cross-subject split from NTU-RGB+D dataset [38] to perform evaluation of these experiments. During testing, a single view (with multiple video clips) is used to perform action recognition. The view-specific classification accuracy scores along with the average is shown in Table 1. We observe that the action recognition performance is relatively better for view 1 in comparison with view 2 and 3. The videos in view 1 are captured at +/- 45 degrees view of actor

Table 2. A comparison of cross-subject (CS) and cross-view (CV) action recognition performance on NTU-RGB+D dataset for RGB modality. RGB-S: using both RGB and skeleton modalities, RGB-DS: using RGB, depth and skeleton modalities.

Method	Modality	Accuracy	
		CS	CV
STA-Hands [1]	RGB-S	73.5	80.2
Pose Est. [28]	RGB-S	84.6	-
DSSCA - SSLM[39]	RGB-DS	74.9	-
CNN-LSTM [27]	RGB	56	-
DA-NET[43]	RGB	-	75.3
Att-LSTM [52]	RGB	63.3	70.6
CNN-BiLSTM [22]	RGB	55.5	49.3
Proposed	RGB	**82.3**	**86.3**

Table 3. A comparison of cross-subject (CS) and cross-view (CV) action recognition performance on NTU-RGB+D dataset for **depth modality**.

Method	Accuracy	
	CS	CV
HOG [32]	32.2	22.3
S-Norm Vector [50]	31.8	13.6
HON4D [33]	30.6	7.3
Shuffle & learn [31]	61.4	53.2
CNN-LSTM [27]	66.2	53.2
CNN-BiLSTM [22]	68.1	63.9
Proposed	**71.8**	**78.7**

and for view 2 and 3 it is either frontal or +/- 90 degrees. However, this is not consistent for M-2 configuration as it never sees the video samples from view 1 during training.

We also observe that when the rendering network is trained for unseen views (M-1 and M-2), it perform better in comparison with seen view prediction (M-3). Predicting unseen query views is relatively difficult for the rendering network and therefore it forces the representation learning network to learn a good representation. Also, the random selection of input views (M-1) allows the network to see different variations in terms of input and query views. Therefore, this configuration (M-1) performs better than M-2 where the input and output views are fixed throughout the training.

4.2 Action Recognition

Based on the above analysis, we select configuration M-1 for rest of our action recognition experiments. For cross-subject experiments, we use two random views as input and render a video from third unseen view. In case of cross-view setup, there are two views (view 2 and 3) available for training. We randomly pick one view as input and the other for rendering. During test time, multiple clips are sampled from the test video for action recognition covering the full length of video. The details of these configurations are shown in Figure 4. The cross-view and cross-subject classification scores on NTU-RGB+D dataset for RGB modality are shown in Table 2. We also evaluate the performance of proposed

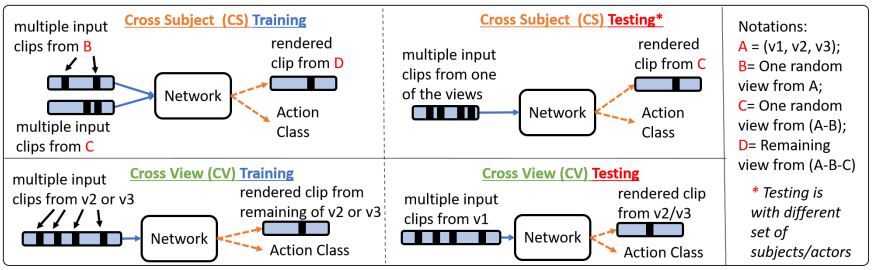

Fig. 4. The details of the training and testing configuration (Strategy M-1) used for cross-view and cross-subject experiments on NTU-RGB+D dataset

Table 4. A comparison of cross-subject (CS) and cross-view (CV) action recognition on N-UCLA MultiviewAction3D dataset.

Method	Modality	Accuracy	
		CS	CV
NKTM [35]	RGB-S	-	75.8
MST-AOG [44]	RGB-S	81.6	73.3
Hanklets [20]	RGB	54.2	45.2
DV-Views[23]	RGB	50.7	58.5
LRCN [6]	RGB	-	64.7
nCTE [11]	RGB	-	68.6
Proposed-scratch	RGB	35.1	43.4
Proposed	RGB	**87.5**	**83.1**

method on depth modality and the evaluation is shown in Table 3. We observe that it performs well on both RGB and depth modality which demonstrates its effectiveness to generalize well across different modalities.

Comparison with state-of-the-art We compare the performance of our proposed method with the recent works on RGB based view-invariant action recognition (Table 2 and Table 3). Our model performs well in both CS and CV evaluation for both RGB and depth modality. We observe that the proposed method provides significant improvement in CV evaluation for both RGB ($\sim 11\%$) and depth ($\sim 15\%$) modality which demonstrates that the learned representation is robust to viewpoint change. Moreover, the performance of our method is comparable to the state-of-the-art approaches employing skeleton modalities.

Model parameters The proposed network use a simple 3D CNN with 7 convolution layers for video encoding. Also, the full network has only around 72M parameters. This is relatively smaller network when compared with existing approaches, such as [22] (ResNet) and [43] (TSN), which utilize a deeper backbone for video encoding.

Table 5. Comparison of classification accuracy to study the effect of cross-view video prediction on NTU-RGB+D dataset. For CS evaluation all the three available views were used for testing whereas for CV evaluation only view 1 was used for testing

Method	Accuracy	
	CS	CV
Baseline	51.4	54.6
Proposed (without prediction)	63.0	65.2
Proposed (seen-view prediction)	71.4	78.1
Proposed (cross-view prediction)	**88.9**	**86.3**

4.3 Transfer Learning for Action Recognition

We use the representation learned on NTU-RGB+D dataset for N-UCLA dataset which has a different set of scenes and users. We perform both CV as well as CS evaluation as suggested in [44] with two different variations. In the first variation, we train the network from scratch and the other one uses transfer learning from NTU-RGB+D dataset. The evaluation of the proposed method on N-UCLA dataset is shown in Table 4. We observe that the network performs poorly when trained from scratch which is due to the small size of this dataset. However, using the learned representations from NTU-RGB+D significantly increases the performance. Our method outperforms previous RGB based approaches [27,22]. This demonstrates the generalization capability of the learned representations across domains.

4.4 Ablations and Discussion

In our previous experiments, we observe that predicting an unseen views helps in learning a better representation for action recognition. We perform some more ablations to study the effect of prediction and effectiveness of BL-NET.

Effect of Rendering: To study the effect of video prediction on the performance of action recognition, we train a network which uses RL-NET for representation learning along with classification without any video prediction. In another variation, we train the proposed network with seen-view prediction. Here, similar setting as before is used except that a random view from input is also selected for prediction. We compare these two baselines to the proposed network with cross-view prediction (M-1). We also use a baseline where the network was trained using a single clip with classification loss. In this experiment, we use the full video length during inference and all available views for CS evaluation. The comparison is shown in Table 5 and we can observe that cross-view prediction provides a significant improvement in the classification scores for both cross-view and cross-subject evaluation.

View-invariant Representation: We compare the representation learned by the proposed RL-NET with autoencoding density models such as Variational Autoencoder (VAE) [16]. The VAE was implemented by replacing the RL-NET model with a CNN network (similar to ENC-NET) and keeping the rest of the network similar to ours. The network was then trained for reconstruction of the

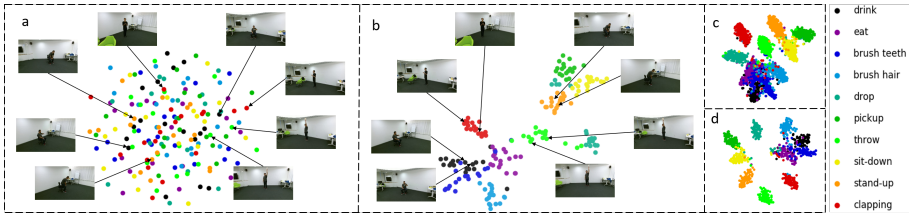

Fig. 5. A comparison of t-SNE visualization of representations learned with: a) Variational Autoencoder (VAE) and b) proposed RL-NET for a subset of 10 activities on NTU-RGB+D dataset. The shown images are the first frame of the video clips. We observe that VAE is indifferent to view awareness of activities and mostly clusters videos with similar visual content. On the other hand, the proposed method is able to cluster activities from different views close to each other even if they have different viewpoints. Effect of multi-view learning: t-SNE visualization of activity representations for a subset of 10 activities on full NTU-RGB+D dataset using: c) *one* input view and d) all *three* views. The learned representation improves with the availability of multiple views using the same network

Table 6. Ablation experiments to study the effect of multiple views during evaluation of CS split in NTU-RGB+D dataset for both RGB and depth modality

Approach	Accuracy	
	RGB	Depth
Single view	82.3	71.8
All views	**88.9**	**79.4**

input video clip along with action classification. We study the 2D t-SNE [29] analysis for the embedding from the last layer of the classifier for a comparison. We observe that the proposed method was able to place the instances from similar classes close to each other despite the change in the viewpoint. VAE on the other hand failed to capture any structure in the representations with varying viewpoints and activity classes. A t-SNE comparison plot of the representation is shown in Figure 5 (a & b).

Effectiveness of BL-NET: The RL-NET performs representation learning based on a set of input observations. These observations can be from different viewpoints and time in a video. The visual appearance of any action changes with viewpoint as well as time. This analogy between time and viewpoint for variation in visual appearance of any action allows us to use the two concepts interchangeably during representation learning. This idea makes the proposed architecture even more powerful as a network trained with some number of views and clips can be tested on different configuration.

Variation in number of input views: We perform an ablation study to validate the robustness of BL-NET to varying number of views during inference. We observe that the performance increases as more number of views are available for representation learning. A comparison is shown in Table 6 for both RGB and depth modality. This is intuitive as different views provide varying perspective

which helps in recognizing the action better. We explored this further with t-SNE visualization of the representations and observe that the action samples are well separated when the representation is learned using multiple views. A comparison is shown in Figure 5 (c & d). The embeddings for activity classes which are confusing (e.g. brush teeth and brush hair) are slightly overlapping with single views (Figure 5c) and are separated with multi-view embedding (Figure 5d).

Variation in number of input clips: The network can also use varying number of input clips to learn a representation. Different action sequences may have varying length and this ability allows the network to see the full action sequence regardless of the video length. During testing we analyzed the effect of number of input clips on the learned representation by introducing more clips than what the network trained for and observed that increasing the input clips leads to a better performance. The action recognition performance increased from 74.6 to 82.3 for CS evaluation on NTU-RGB+D when we increase the number of clips from 6 to full video length. This also validates the robustness of BL-NET for varying length action sequences. The proposed network can also generalize well to testing with single view videos irrespective of how it was trained. For example, the CS network for NTU-RGB+D is trained using 2 input views, but it still performs well in action recognition using only single view clips.

5 Conclusion

In this work, we propose a novel unsupervised deep learning framework for action recognition in a multi-view environment. The proposed framework can be trained end-to-end without the need of any pre-trained weights. We demonstrate the effectiveness of the proposed approach for the task of cross-view as well as cross-subject action recognition on multiple datasets. The proposed approach is effective with RGB videos and we also validate the generalization capability of the proposed framework for depth modality. The framework is adaptive for number of input views which makes it suitable for multi-view as well as novel-view action recognition. The generalization capability and its adaptive nature makes it useful for other problem domains in a multi-view environment.

Acknowledgement This research is based upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via IARPA R&D Contract No. D17PC00345. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

1. Baradel, F., Wolf, C., Mille, J.: Human action recognition: Pose-based attention draws focus to hands. In: The IEEE ICCV Workshops (Oct 2017)
2. Ben Tanfous, A., Drira, H., Ben Amor, B.: Coding kendall's shape trajectories for 3d action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2840–2849 (2018)
3. Byeon, W., Wang, Q., Srivastava, R.K., Koumoutsakos, P., Vlachas, P., Wan, Z., Sapsis, T., Raue, F., Palacio, S., Breuel, T., et al.: Contextvp: Fully context-aware video prediction. In: Proceedings of the IEEE CVPR Workshops (2018)
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR (2017)
5. Clark, A., Donahue, J., Simonyan, K.: Efficient video generation on complex datasets. arXiv preprint arXiv:1907.06571 (2019)
6. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: IEEE conference on CVPR (2015)
7. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1110–1118 (2015)
8. Eslami, S.A., Rezende, D.J., Besse, F., Viola, F., Morcos, A.S., Garnelo, M., Ruderman, A., Rusu, A.A., Danihelka, I., Gregor, K., et al.: Neural scene representation and rendering. Science (2018)
9. Fernando, B., Bilen, H., Gavves, E., Gould, S.: Self-supervised video representation learning with odd-one-out networks. In: IEEE conference on CVPR (2017)
10. Goyal, P., Hu, Z., Liang, X., Wang, C., Xing, E.P., Mellon, C.: Nonparametric variational auto-encoders for hierarchical representation learning. In: ICCV. pp. 5104–5112 (2017)
11. Gupta, A., Martinez, J., Little, J.J., Woodham, R.J.: 3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2601–2608 (2014)
12. Hochreiter, S., Schmidhuber, J.: Lstm can solve hard long time lag problems. In: NeurIPS (1997)
13. Isik, L., Tacchetti, A., Poggio, T.A.: A fast, invariant representation for human action in the visual system. Journal of neurophysiology (2017)
14. Jayaraman, D., Gao, R., Grauman, K.: Shapecodes: self-supervised feature learning by lifting views to viewgrids. In: European conference on computer vision (2018)
15. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: A new representation of skeleton sequences for 3d action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3288–3297 (2017)
16. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
17. Lakhal, M.I., Lanz, O., Cavallaro, A.: View-lstm: Novel-view video synthesis through view decomposition. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
18. Ledig, C., Theis, L., Huszár, F., Caballero, J., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE conference on CVPR (2017)

19. Lee, H.Y., Huang, J.B., Singh, M., Yang, M.H.: Unsupervised representation learning by sorting sequences. In: International Conference on Computer Vision (ICCV) (2017)
20. Li, B., Camps, O.I., Sznaier, M.: Cross-view activity recognition using hankelets. In: IEEE CVPR (2012)
21. Li, C., Cui, Z., Zheng, W., Xu, C., Yang, J.: Spatio-temporal graph convolution for skeleton based action recognition. In: AAAI Conference on Artificial Intelligence (2018)
22. Li, J., Wong, Y., Zhao, Q., Kankanhalli, M.: Unsupervised learning of view-invariant action representations. In: Advances in Neural Information Processing Systems (2018)
23. Li, R., Zickler, T.: Discriminative virtual views for cross-view action recognition. In: IEEE CVPR (2012)
24. Liu, J., Shahroudy, A., Xu, D., Kot, A.C., Wang, G.: Skeleton-based action recognition using spatio-temporal lstm network with trust gates. IEEE transactions on pattern analysis and machine intelligence **40**(12), 3007–3021 (2017)
25. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal lstm with trust gates for 3d human action recognition. In: European Conference on Computer Vision. pp. 816–833. Springer (2016)
26. Liu, M., Liu, H., Chen, C.: Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition **68**, 346–362 (2017)
27. Luo, Z., Peng, B., Huang, D.A., Alahi, A., Fei-Fei, L.: Unsupervised learning of long-term motion dynamics for videos. In: IEEE Conference on CVPR (2017)
28. Luvizon, D.C., Picard, D., Tabia, H.: 2d/3d pose estimation and action recognition using multitask deep learning. In: IEEE Conference on CVPR (2018)
29. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
30. Mathieu, M., Couprie, C., LeCun, Y.: Deep multi-scale video prediction beyond mean square error. ICLR (2016)
31. Misra, I., Zitnick, C.L., Hebert, M.: Shuffle and learn: unsupervised learning using temporal order verification. In: European Conference on Computer Vision. Springer (2016)
32. Ohn-Bar, E., Trivedi, M.: Joint angles similarities and hog2 for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 465–470 (2013)
33. Oreifej, O., Liu, Z.: Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 716–723 (2013)
34. Rahmani, H., Mahmood, A., Huynh, D., Mian, A.: Histogram of oriented principal components for cross-view action recognition. IEEE transactions on PAMI (2016)
35. Rahmani, H., Mian, A.: Learning a non-linear knowledge transfer model for cross-view action recognition. In: Proceedings of the IEEE conference on CVPR (2015)
36. Regmi, K., Borji, A.: Cross-view image synthesis using conditional gans. In: IEEE Conference on CVPR (2018)
37. Saito, M., Matsumoto, E., Saito, S.: Temporal generative adversarial nets with singular value clipping. In: IEEE International Conference on Computer Vision (ICCV) (2017)
38. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In: Proceedings of the IEEE conference on CVPR (2016)

39. Shahroudy, A., Ng, T.T., Gong, Y., Wang, G.: Deep multimodal feature analysis for action recognition in rgb+ d videos. IEEE Transactions on PAMI (2018)
40. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: International conference on machine learning. pp. 843–852 (2015)
41. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. arXiv preprint arXiv:1707.04993 (2017)
42. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: NeurIPS (2016)
43. Wang, D., Ouyang, W., Li, W., Xu, D.: Dividing and aggregating network for multi-view action recognition. In: The European Conference on Computer Vision (ECCV) (September 2018)
44. Wang, J., Nie, X., Xia, Y., Wu, Y., Zhu, S.C.: Cross-view action modeling, learning and recognition. In: IEEE Conference on CVPR (2014)
45. Wang, X., Gupta, A.: Unsupervised learning of visual representations using videos. In: IEEE ICCV (2015)
46. Wen, Y.H., Gao, L., Fu, H., et al.: Graph cnns with motif and variable temporal block for skeleton-based action recognition. In: AAAI Conference on Artificial Intelligence (2019)
47. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500 (2017)
48. Xu, X., Chen, Y.C., Jia, J.: View independent generative adversarial network for novel view synthesis. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
49. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: AAAI Conference on Artificial Intelligence (2018)
50. Yang, X., Tian, Y.: Super normal vector for activity recognition using depth sequences. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 804–811 (2014)
51. Zhang, P., Lan, C., Xing, J., Zeng, W., Xue, J., Zheng, N.: View adaptive neural networks for high performance skeleton-based human action recognition. IEEE PAMI (2019)
52. Zhang, P., Xue, J., Lan, C., Zeng, W., Gao, Z., Zheng, N.: Adding attentiveness to the neurons in recurrent neural networks. In: European Conference on Computer Vision (2018)