Deep Attention Network for Egocentric Action Recognition

Minlong Lu, Ze-Nian Li, Yueming Wang, and Gang Pan

***Abstract*—Recognizing camera wearer's actions from videos captured by an egocentric camera is a challenging task. In this work, we employ a two-stream deep neural network composed of an appearance-based stream and a motion-based stream to recognize egocentric actions. Based on the insight that human action and gaze behavior are highly coordinated in object manipulation tasks, we propose a spatial attention network to predict human gaze in the form of attention map. The attention map helps each of the two streams to focus on the most relevant spatial region of the video frames to predict actions. To better model the temporal structure of the videos, a temporal network is proposed. The temporal network incorporates bi-directional long short-term memory (LSTM) to model the long-range dependencies to recognize egocentric actions. The experimental results demonstrate that our method is able to predict attention maps that are consistent with human attention, and achieve competitive action recognition performance with the state-of-the-art methods on the GTEA Gaze and GTEA Gaze+ datasets.**

***Index Terms*—Attention Network, Top-down Attention, Egocentric Vision, Action Recognition, LSTM.**

I. Introduction

Understanding human behavior from videos has been a highly active research topic in computer vision. With the availability of various wearable cameras, there is a growing interest in using first-person videos to understand the camera wearer's behavior. The wearable camera is usually mounted on a person's head and its optical axis is aligned with the person's field of view. It can even be mounted on an animal's head for building a vision-augmented cyborg intelligent system [1], [2]. Recognizing actions using first-person videos, or egocentric videos, is different from that using third-person videos. This is because the camera wearer's poses are mostly unavailable in these videos. And unlike third person videos where the camera is either static or moving smoothly, strong motions are commonly present in egocentric videos due to the head motion of the camera wearer. These aspects make egocentric action recognition very challenging.

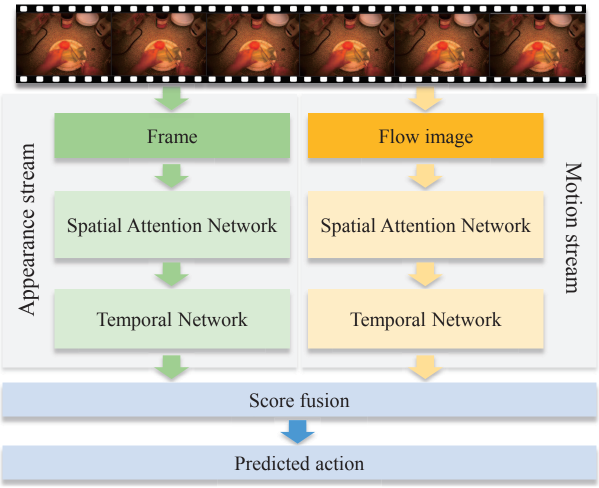


Fig. 1. Approach overview. The spatial attention networks predict attention maps to select relevant regions to focus on. The temporal networks model the forward and backward information for action recognition.

Researchers have explored a rich set of visual features, including object-centric features and egocentric cues for action recognition in first-person videos. The object features aim at capturing the appearance changes of objects and are shown to be effective in characterizing egocentric actions [3], [4]. The egocentric cues include the first person's head/hand motion and hand pose [5], [6], which can reveal the underlying actions of the camera wearer and are shown to be complementary to object-centric representations. Recent works have attempted to employ the feature learning capability of convolutional neural networks (CNNs) [7] for egocentric action recognition and have achieved good performance [8], [9]. In order to directly incorporate object and egocentric cues, these models use preprocessed inputs such as hand mask, homography [9], and localized objects [8].

Using eye-tracking devices, the gaze or eye fixation of the person can be recorded while interacting with the physical world, which are also utilized to facilitate egocentric action recognition during object manipulation tasks [5], [10]. The eye movements reflects a person's thinking process and represents human attention [11], which can be divided into two categories: bottom-up attention and top-down attention [12], [13]. The bottom-up attention includes the human attention when they are performing free-viewing on a scene or an image, in which the objects or regions that "stand out" relative to the neighboring parts (saliency) attract human attention. In comparison, the human attention when they are performing

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under the Grant RGP36726, National Key Research and Development Program of China (2017YFB1002503), Natural Science Foundation of China (No.61772460), and Zhejiang Provincial Natural Science Foundation of China (LR15F020001). Corresponding author: G. Pan and M. Lu.

M. Lu is with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China, and also with the School of Computing Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada.

Z.N. Li is with the School of Computing Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada.

Y. Wang is with the Qiushi Academy for Advanced Studies, Zhejiang University, Hangzhou 310027, China.

G. Pan is with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China.

certain tasks (e.g. object manipulation) belongs to the category of top-down attention, which is task-driven. The human eye movements and fixation during performing these tasks are very different from during free-viewing [11]. It is demonstrated in [14] that a substantial percentage of human eye fixations falls on the task-relevant regions during object manipulation tasks, which require hand-eye coordination. For example, when pouring water into a bottle, instead of paying attention to the salient objects, the person has to fixate on the opening of the bottle and also monitor the water level in the bottle. In these tasks, the point of eye fixation may not be at the location which is the most visually salient (bottom-up attention), but rather will correspond to the most relevant location depending on the task demands and human action (top-down attention).

Based on these insights, we propose an attention-based deep network that exploit the spatial and temporal structure of the egocentric videos for action recognition. Our model has a two-stream architecture which consists of an appearance stream and a motion stream, as shown in Figure 1. A spatial attention network is incorporated in each stream to learn human attention using gaze as ground truth. It shares the convolutional layers in the stream and has a separate branch to predict a human attention map. This attention map helps our model to focus on the most relevant spatial region of the inputs to recognize egocentric actions. The temporal network incorporates bi-directional long short-term memory (LSTM) to model the long-range dependencies of the video frames.

Our contribution can be summarized as follows: (1) We propose a spatial attention network and a temporal network, which are incorporated in a two-stream architecture for egocentric action recognition. Our model achieves state-of-the-art performance on GTEA Gaze dataset. (2) We provide detailed ablation analysis to demonstrate how the proposed spatial attention network and temporal network contribute to the overall performance. (3) To the best of our knowledge, our method is the first successful deep network-based method that models human gaze behavior and top-down attention. By comparing to a prediction-oriented attention model and a saliency-based model, we demonstrate both quantitatively and qualitatively that our spatial attention network with gaze supervision is capable of learning better attention mechanism for egocentric action recognition.

II. Related Work

A. Action Recognition

Action recognition from a third person view has been one of the key problems in computer vision [15]. A large family of methods are based on high-dimensional encodings of local features, such as histogram of oriented gradients (HOG) [16], histogram of flow (HoF) [17] and motion boundary histograms (MBH) [18]. These features are usually extracted from Space-Time Interest Point (STIP) [19] or along dense trajectories (DT) [20], and can be encoded into the bag of word (BoW) representation for action recognition.

There are also a number of attempts to develop deep neural networks for action recognition [21], [22], [23]. A two-stream architecture is proposed in [24], which feeds video frames and optical flow images into separate CNN streams and the scores of the two streams are fused for the prediction. Temporal models such as long short-term memory (LSTM) [25] are employed on top of the CNNs for modeling long-range temporal dependencies for action recognition in videos, which achieves good performance [26], [27], [28]. The convolution and pooling operations of CNNs are extended to 3D in models such as C3D [29] and I3D [30], which learn spatiotemporal features from the videos. The 3D operations are factorized into separate spatial and temporal components in [31] to facilitate optimization.

B. Egocentric Vision

There have been several recent advances in egocentric vision, such as video summarization [32], video stabilization [33], object recognition [34] and action recognition [35], [4]. In egocentric action recognition, researchers have found that traditional spatial-temporal features do not work well due to the camera motion [3], [36]. With the help of motion compensation, large improvement can be achieved using these features [6]. Object-centric features [4] are used to capture the appearance changes of objects, and egocentric cues based on head movement, hand pose and gaze are proposed for better characterizing egocentric actions [6].

Recent works have attempted to employ CNNs to tackle egocentric action recognition problem [8], [9], [37]. To directly incorporate egocentric cues, an Ego ConvNet is proposed to train on stacked input of hand mask, homography image, and saliency maps [9]. The ego stream is then fused with the other two streams for the final prediction. In [8], networks are trained to segment hand and localize object, and then the object of manipulation is cropped as the input to the appearance stream. The appearance and motion streams are fused by a fully-connected layer to recognize the objects and action verbs jointly. In spite of the state-of-the-art performance, its architecture is customized for classifying action categories with “verb+object” form and requires additional annotations such as hand masks, object locations and object labels.

C. Attention Model

Attention models have been proved successful in variant vision tasks, such as object recognition [38], image and video captioning [39], [40], action recognition [41], [42], [43], and visual question answering [44], [45]. The visual attention mechanism aims to identify interesting regions in the visual data and focus on these regions to extract relevant information, which mimics the human perception and thinking process during accomplishing certain tasks.

In the attention models, a probability distribution over a grid of features is first predicted to indicate the level of attention on each region. The soft attention models use the attention distribution to re-weight the features, while the hard attention models select the feature with the highest probability to represent the data. Both soft and hard attention models are explored in [40] for generating image captions. The soft attention model is trained using back-propagation and the hard attention model is trained using reinforcement algorithm.

Attention mechanism is extended to temporal domain in [39], [46], where the models learn to select relevant video segments for video description and action recognition. Based on the insight that image question answering requires multiple steps of reasoning, stacked attention networks are proposed in [45] to progressively focus on different regions of the image to infer the answer. The spatial transformer networks [47] introduce affine transformations to the CNN feature maps, which allows the model to attend to arbitrary regions of the data.

These attention models learn to select the most relevant part of the data for the task implicitly. With the availability of human gaze information in egocentric videos, we are able to use this real human attention to train our spatial attention model in a supervised way. To the best of our knowledge, our method is the first attempt to use deep spatial attention models in egocentric action recognition, which models gaze behavior and the task-dependent top-down human attention.

III. The Proposed Method

In this work, we employ a two-stream architecture composed of an appearance stream and a motion stream to recognize egocentric actions. The spatial attention network is incorporated in each stream to predict human attention distribution using gaze information as ground truth. The attention distribution helps our model to selectively focus on the most relevant part of the data to predict actions. The temporal network incorporates bi-directional LSTM to model the long range temporal structure of the videos for recognizing actions. In this section, we will describe our spatial and temporal networks, and provide detailed framework of our two-stream architecture.

A. Spatial Attention Network

Our spatial attention network takes the feature map of the last convolutional layer in the generic CNNs as input. We denote the feature map as $X \in \mathbf{R}^{K \times K \times D}$, where K is the spatial resolution of the feature map and D is the number of the feature channels. We feed the feature map to convolutional layers to predict an attention distribution $A \in \mathbf{R}^{K \times K}$ over the grid of features as:

$$A = f(X \otimes w + b), \tag{1}$$

where w is the convolution kernel and b is the bias term.

The previous attention models do not have direct supervision on this predicted attention distribution [40], [41], [45]. Instead, they use the distribution to either weighted average the features or select the features with highest attention, then the model is trained in a prediction oriented manner by attempting to minimize the prediction error of the final task. Therefore, these attention models implicitly learns an attention mechanism to focus on certain regions of the input that in favor of the final prediction.

In this work, we utilize the human gaze information as ground-truth to enforce the training of our spatial attention network, see Fig 2. Therefore our model is able to learn top-down task dependent human attention to select the relevant regions for better action recognition. The eye movement can be tracked using wearable tracking system such as Tobii eye tracking glasses, and then synchronized with the egocentric videos. The eye fixation position in the scene is then transformed to the image coordinate and represented as the gaze location (x, y) in each frame. We generate the ground-truth human attention distribution $A^{gt} \in \mathbf{R}^{K \times K}$ by applying a Gaussian bump on the gaze location. Each entry a_{ij}^{gt} in A^{gt} represents attention weight and is computed by:

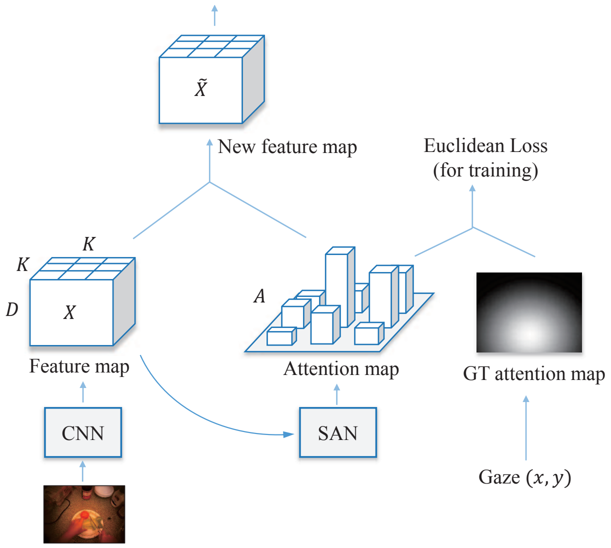


Fig. 2. The spatial attention network (SAN), which takes the original feature map as input and produces an attention map. The SAN is trained using Euclidean loss which compares the predicted attention map with ground truth (GT) attention map generated using human gaze location.

$$a_{ij}^{gt} = e^{-\frac{(i-x')^2+(j-y')^2}{2\sigma^2}}, \tag{2}$$

where $i, j \in [1, K]$ are the spatial indexes in the attention map, x', y' are the scaled gaze coordinates in the interval of $(0, K)$, and σ is set to be $K/2$. Euclidean loss is used to train the spatial attention network during error back-propagation, which measures the prediction error of our attention distribution as:

$$L = \frac{1}{2K^2} \sum_{i=1}^{K} \sum_{j=1}^{K} (a_{ij} - a_{ij}^{gt})^2, \tag{3}$$

where a_{ij} is the predicted attention weight in A. Based on the attention distribution, we re-weight the feature map X to produce a new feature map $\tilde{X} \in \mathbf{R}^{K \times K \times D}$ by:

$$\tilde{X}_{ij} = a_{ij} \cdot X_{ij}. \tag{4}$$

$\tilde{X}_{ij}, X_{ij} \in \mathbf{R}^D$ are the corresponding feature vectors of $\tilde{X}$ and X at position (i, j), respectively. This attention mechanism constructs a more informative feature map $\tilde{X}$, since higher weight can be assigned to visual regions that are more relevant to the current action. The feature map $\tilde{X}$ is then processed by the consecutive fully connected layers to produce a feature vector $\mathbf{x}$. The ground-truth gaze is only used during the training phase for the spatial attention network to learn the human attention. At test time, our model receives only the visual input and predicts both attention and action.

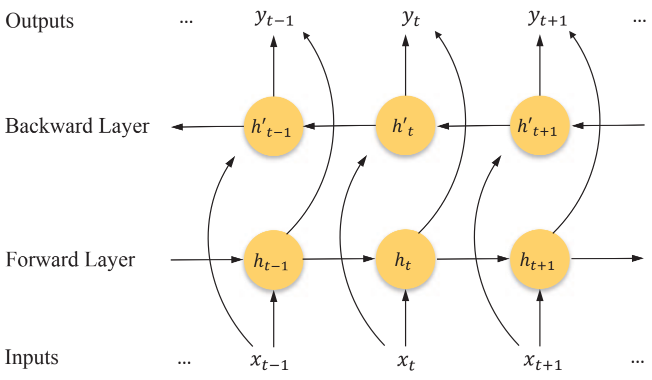


Fig. 3. The temporal network: bi-direction LSTM.

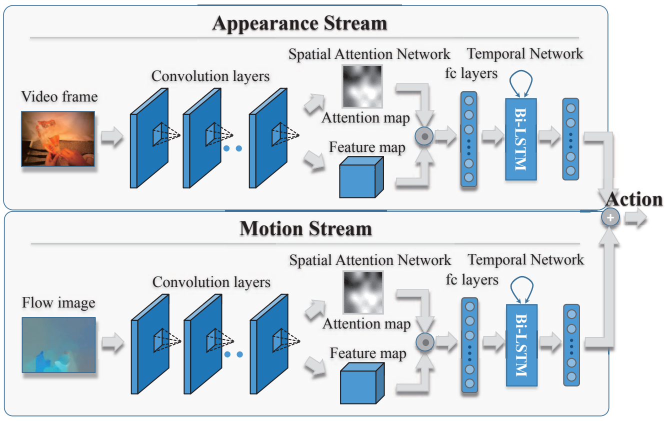


Fig. 4. Framework details of our two-stream model.

B. Temporal Network

In the previous deep models, stacked optical flow is used to model the temporal dynamics of the videos [24], [8]. In order to better exploit the temporal structure of egocentric videos for action recognition, we incorporated a bi-directional LSTM in our temporal network.

Long short-term memory (LSTM) [25] is stable and powerful for modeling long-range temporal dependencies without the vanishing gradient problem of the simple RNNs. Its innovation is the introduction of the "memory cell" c_t to accumulate the state information. The cell is accessed, written and cleared by several controlling gates, which enables LSTM to selectively forget its previous memory states and learn long-term dynamics. Given x_t as the input of an LSTM cell at time t, the cell activation can be formulated as:

$$\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \\
g_t &= \phi(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \phi(c_t),
\end{aligned} \tag{5}$$

where σ stands for the sigmoid function, ϕ stands for the tanh function, and $\odot$ denotes the element-wise multiplication. In addition to the hidden state h_t and memory cell c_t, LSTM has four controlling gates: i_t, f_t, o_t, and g_t, which are the input, forget, output, and input modulation gate respectively. The input gate i_t controls what information in g_t to be accumulated into the cell c_t. While the forget gate f_t helps the c_t to maintain and selectively forget information in previous state c_{t-1}. Whether the updated cell state c_t will be propagated to the final hidden state h_t representation is controlled by the output gate o_t.

The overall structure of the memory cell and the regulating gates make LSTM suitable for modeling complex temporal relationships that may span a long range. However, one shortcoming of the conventional LSTM is that it is only able to make use of previous context. We incorporate the bi-directional LSTM to process the data in both directions with two separate hidden layers, see Fig 3. In addition to the forward LSTM layer that produces a sequence of hidden states h_t, we have a backward LSTM layer that produces a sequence of hidden states h'_t based on the information of the next time step $t+1$:

$$\begin{aligned}
i'_t &= \sigma(W'_{xi}x_t + W'_{hi}h'_{t+1} + b'_i) \\
f'_t &= \sigma(W'_{xf}x_t + W'_{hf}h'_{t+1} + b'_f) \\
o'_t &= \sigma(W'_{xo}x_t + W'_{ho}h'_{t+1} + b'_o) \\
g'_t &= \phi(W'_{xc}x_t + W'_{hc}h'_{t+1} + b'_c) \\
c'_t &= f'_t \odot c'_{t+1} + i'_t \odot g'_t \\
h'_t &= o'_t \odot \phi(c'_t).
\end{aligned} \tag{6}$$

The hidden states of the two LSTM layers are combined to produce the output by:

$$y_t = W_{hy}h_t + W'_{hy}h'_t + b_y, \tag{7}$$

where W_{hy} and W'_{hy} are the weight matrices and b_y is the bias term. The number of elementary operations of our bi-directional LSTM is linear in terms of the number of parameters N, which is the number of entries in the matrices Ws. Assume that X, H, Y are the dimensions of the input vector x, hidden state h, and output vector y respectively, N is proportional to $XH + HH + HY$. Our temporal network takes a consecutive T frames as input to model their temporal structure, therefore the time complexity for each iteration is $O(TXH + THH + THY)$.

C. Two-stream Architecture

It has been proved successful to decompose videos into spatial and temporal components for action recognition [24], [26]. The spatial component, in the form of raw frame appearance, contains information about scenes and objects depicted in the video. The temporal component, in the form of motion across the frames, conveys the motion of the camera and objects in the scene. Our two-stream architecture follows the successful practices in action recognition [24], [26]. The appearance stream and motion stream in our model coupe with the spatial and temporal video components respectively. Due to the complementary feature learned from the inputs, the fused network is able to produce improved performance.

Each stream of our model contains a generic CNN to extract feature maps from the spatial and temporal components. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2901707, IEEE Transactions on Image Processing

CNN in the appearance stream operates on raw video frames. It captures appearance features such as hand-object configuration and object attributes. The CNN in the motion stream takes optical flow images as input and learns complementary features. We adopt the flow image encoding method in [26]. The first two channels of the flow image are computed by centering x and y flow values around 128 and scaling the values to fall between 0 and 255. The third channel of the flow image is created by computing the flow magnitude. This flow image models local temporal structure and conveys short term information about the camera, hand or object motion.

The advantage of decoupling the appearance and motion streams is that it allows us to utilize the large amount of annotated image data (e.g. ImageNet) for pre-training each of the CNN. The CNNs extract discriminative feature maps, which are then used by our spatial attention and temporal networks to recognize egocentric actions, as illustrated in Fig 4. The softmax scores of the two streams are fused to predict the action labels.

IV. Experiments

A. Datasets and Experimental Setup

We evaluate our proposed method on two public datasets: GTEA Gaze (Gaze) and GTEA Gaze+ (Gaze+). These datasets are collected using a head-mounted camera and the activities performed by the camera wearer involve hand-object interactions. These datasets include action annotations as well as gaze information. Each action is represented by a verb and a set of nouns, for example “put lettuce (on) plate”. The gaze information is represented as a coordinate in the video frame, indicating the location where the person is looking at. The details of the datasets are introduced below.

- **Gaze dataset** [5] contains 17 video sequences performed by 14 different subjects. The total duration of these videos is one hour with frame rate 15fps, and the frame resolution is 640×480. The gaze data is obtained using Tobii eye tracking glasses. Previous works usually report results on fix splits on this dataset, where 13 sequences are used as training data and 4 sequences are used as testing data. There are 40 action categories and a total of 331 action instances.
- **Gaze+ dataset** [5] contains 37 video sequences performed by 6 different subjects. The total duration of these videos is 9 hours with frame rate 24fps, and the frame resolution is 1280×960. The gaze data is obtained using SMI eye tracking glasses. Previous works usually report leave-one-subject-out cross validation accuracy on this dataset. There are 44 action categories and a total of 1958 action instances.

Each action instance in these datasets is a trimmed video segment with tens to hundreds of consecutive frames, during which the camera wearer completes one action. Therefore each instance and all the frames it contains have a single action label. We use the instance level accuracy to evaluate the performance of our method, which is the percentage of the instances that are classified correctly in the testing set.

It is non-trivial to determine how many convolution layers are needed for our spatial attention network to learn the human attention. Therefore we test several architectures with different numbers of convolution layers (more details in Section IV-F). Based on the results, we choose to use spatial attention network with one convolution layer in our experiments.

Our temporal network takes a set of consecutive frames as input (16 as suggested in [26]), and use the bi-directional LSTM to learn the forward and backward dependencies among the frames. During each training iteration, we randomly select a start frame for an instance and use the consecutive 16 frames to train the model. At test time, we extract 16 frame clips with a stride of 8 frames from each test instance to produce the frame scores. The scores of the overlapped 8 frames are averaged. After evaluating all the 16 frame clips, the scores of all the frames in the instance are averaged to predict the label of this instance. The experimental settings of LSTM training and testing are adopted from [26].

B. Implementation Details

We implement our model using Caffe [48], and we adopt the LSTM implementation in [26]. The VGG net [49] 16 layer architecture is adopted as the generic CNN in our appearance and motion streams. We utilize the network weights pre-trained on the ImageNet [50] dataset to initialize our CNNs and then train the networks on the Gaze and Gaze+ datasets.

Leave-one-subject-out cross-validation is performed by the comparing methods on Gaze+ dataset. However, this is very computationally intensive for deep learning frameworks and makes hyper-parameter tuning challenging. In our experiment, we choose our parameters using the first subject as validation set on Gaze+ dataset, and fix parameters for the rest of the dataset as well as for the Gaze dataset. We use 16 timesteps and batch size 8 for the bi-directional LSTM layers, and the LSTM has hidden vector with a dimension of 1024.

Similar to the training procedure used in [26], we first train the generic CNNs with a base learning rate of 0.001 and decrease it by a factor of 0.1 for every 3000 iterations until the maximum iteration 30000. Then we add the spatial attention network and train the model by fixing the CNN convolution kernels. The base learning rate is set to be 10^{-6} and the same learning rate decreasing criteria is used. After that, the temporal network is added and trained for 10000 iterations with the earlier components fixed. The base learning rate of this step is 0.01 and the learning rate decreases in the same way. Finally, joint training for each stream is performed with base learning rate of 10^{-4} and the same learn rate decrease policy. The advantage of this training procedure is that it ensures the convergence of each component, and we do not need to worry about the learning speed and loss magnitude differences between them.

C. Comparison with Previous Methods

We compare our method with well-performing methods using traditional features such as DT [18] and improved DT (IDT) [51], as well as the best-known methods proposed in [6], [8]. Li *et al.* [6] provide a systematic evaluation of different

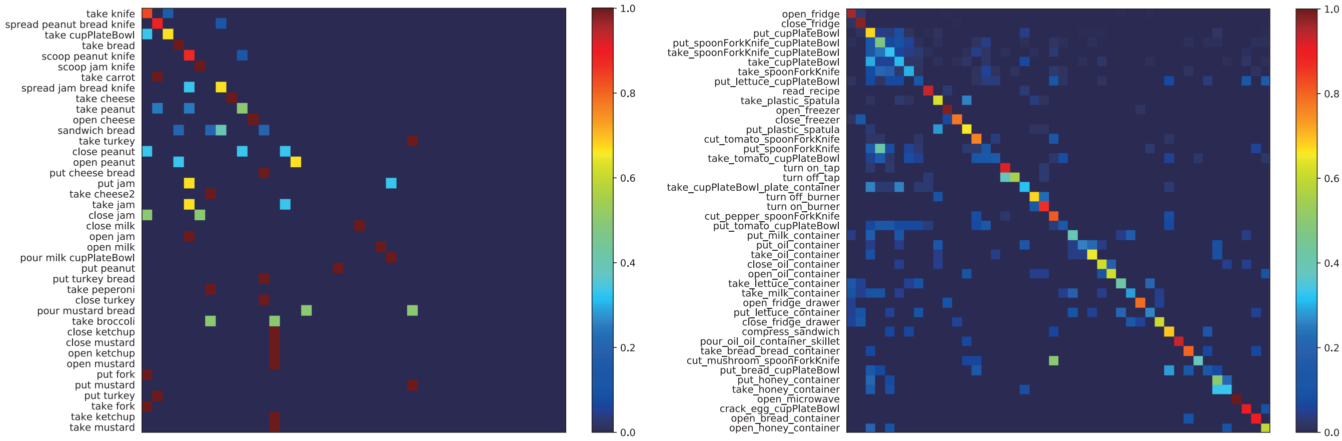


Fig. 5. Confusion matrices of our method on Gaze (left) and Gaze+ (right) datasets. Action categories are sorted based on decreasing number of instances as in [6].

TABLE I
COMPARISON OF THE ACTION RECOGNITION ACCURACY OF OUR METHOD WITH PREVIOUS METHODS.

Methods	Gaze (40)	Gaze+ (44)
Wang *et al.*: DT [18]	34.10	42.40
Wang *et al.*: IDT [51]	27.70	49.60
Li *et al.*: O+E+H [6]	35.10	57.40
Li *et al.*: O+M+E+H [6]	35.70	60.50
Li *et al.*: O+M+E+G [6]	39.60	60.30
Ma *et al.*: object-cnn [8]	35.56	46.38
Ma *et al.*: motion+object-svm [8]	16.00	34.75
Ma *et al.*: motion+object-joint [8]	43.42	66.40
Ours: appearance stream	42.86	57.63
Ours: motion stream	37.36	57.42
Ours: two-stream fusion	48.35	64.74

combinations of features for egocentric action recognition, and the best performing ones are included in Table I. **O** stands for object features obtained by concatenating the Fisher Vectors (FVs) from HoG, LAB color histogram and LBP; **M** stands for motion features, which are a concatenation of the FVs from trajectory features, MBHx, MBHy and HoF; **E** stands for egocentric features which are computed by concatenating FVs from head motion and manipulation point; **G** and **H** represents the selected local descriptors near the gaze point and the manipulation point estimated by hand shape. Ma *et al.* [8] propose a deep architecture that contains an object CNN and a motion CNN to recognize actions.

The action recognition accuracies of our appearance stream, motion stream, and the overall two-stream model are shown in Table I. The comparing results are taken from [6], [8]. It can be seen that score fusion can result in a significant improvement over both the appearance stream and the motion stream. This demonstrates that the two streams learn complementary information for action recognition. Our method consistently outperforms the DT [18], IDT [51] methods, as well as the top 3 feature combinations proposed in [6] on both the Gaze and Gaze+ datasets. The confusion matrices of our method are shown in Fig 5. The action categories are sorted with the decreasing number of instances. Our method can get most of the categories correct on Gaze+ dataset. The deviation of the last few categories in Gaze dataset is due to the imbalanced data distribution in this preliminary dataset [6]. There are limited training and testing instances (1-2) for those categories and misclassifying one instance can result in a large penalty in the confusion matrix, similar problem is also discussed in [6].

The method in [8] performs action recognition with the help of an explicitly designed object recognition network, which requires additional annotations of hand masks, object locations and object labels. It first utilizes a fully convolutional network to segment hand masks, based on which it fine-tunes an object localization network to detect the objects being manipulated in egocentric videos. The cropped object images are used as input to the object recognition network, which is combined with the motion network for joint action recognition. Since an action is defined as “verb+object”, this object network provides additional information for recognizing the action. Therefore the method in [8] is not directly comparable to our model, and we would have also benefited if we were to utilize the object recognition results. However, our model is able to achieve comparable performance on the two datasets. Besides, our model has a simple and general architecture, while the method in [8] has a customized architecture and complex pipeline.

D. Ablation Study

To analyze the performance of the spatial attention network and the temporal network in our two-stream architecture, we conduct a detailed ablation study by testing each component in each stream and their combinations to see how they contribute to the action recognition accuracy. The ablation study is conducted on the Gaze+ dataset. Each stream contains a generic CNN, a spatial attention network and a temporal network. We test the combination of these components, which results in the models listed below:

1. **RGB-o**: this model contains only the generic CNN of the appearance stream.
2. **RGB-s**: this model contains the generic CNN and the spatial attention network of the appearance stream.

TABLE II
DETAILED ABLATION STUDY OF OUR METHOD ON GAZE+ DATASET. THERE ARE 6 SUBJECTS IN THIS DATASET AND WE PRODUCE ACTION RECOGNITION ACCURACIES ON EACH OF THE SUBJECTS AND COMPUTE THE AVERAGE.

Methods	RGB-o	RGB-s	RGB-t	RGB-st	Flow-o	Flow-s	Flow-t	Flow-st	Fuse-o	Fuse-s	Fuse-t	Fuse-st
Subject #1	49.85	51.65	51.45	51.65	50.83	52.27	51.85	53.10	54.55	57.02	56.61	58.26
Subject #2	72.26	72.63	71.17	71.17	67.15	67.88	68.24	66.42	75.18	75.18	73.72	76.28
Subject #3	51.22	51.00	51.88	52.77	51.00	53.44	51.44	52.11	55.65	56.98	55.65	58.31
Subject #4	62.00	62.29	61.71	61.43	66.29	64.57	64.57	65.14	68.86	71.42	70.57	72.57
Subject #5	57.64	61.81	59.03	59.03	52.78	50.00	53.47	52.08	57.64	62.50	61.81	61.81
Subject #6	52.74	55.27	55.70	55.70	56.96	56.96	54.85	57.81	63.71	64.97	67.09	67.09
Average	56.39	57.42	57.27	57.63	56.86	57.37	57.37	57.42	61.65	63.56	62.99	64.74

3. **RGB-t**: this model contains the generic CNN and the temporal network of the appearance stream.
4. **RGB-st**: this is the appearance stream of our method, which contains the generic CNN, the spatial attention network and the temporal network.
5. **Flow-o**: this model contains only the generic CNN of the motion stream.
6. **Flow-s**: this model contains the generic CNN and the spatial attention network of the motion stream.
7. **Flow-t**: this model contains the generic CNN and the temporal network of the motion stream.
8. **Flow-st**: this is the motion stream of our method, which contains the generic CNN, the spatial attention network and the temporal network.
9. **Fuse-o**: the score fusion from RGB-o and Flow-o.
10. **Fuse-s**: the score fusion from RGB-s and Flow-s.
11. **Fuse-t**: the score fusion from RGB-t and Flow-t.
12. **Fuse-st**: the score fusion from RGB-st and Flow-st, which is our two-stream model.

The detailed results of these models are listed in Table II. We can see that both the “-s” and “-t” models are able to outperform the “-o” models, and the “-st” models achieve the best results. This demonstrates that our spatial attention network and temporal network are able to better model the video structures for action recognition. By including each component, the performance of our model increases consistently. The results of different fusion combinations of the corresponding RGB and Flow models are shown in columns 10-13 in Table II. The score fusion can result in the improvement of accuracy over each of the individual stream. The best performance is achieved by the Fuse-st model, which fuses the score of our appearance stream and motion stream.

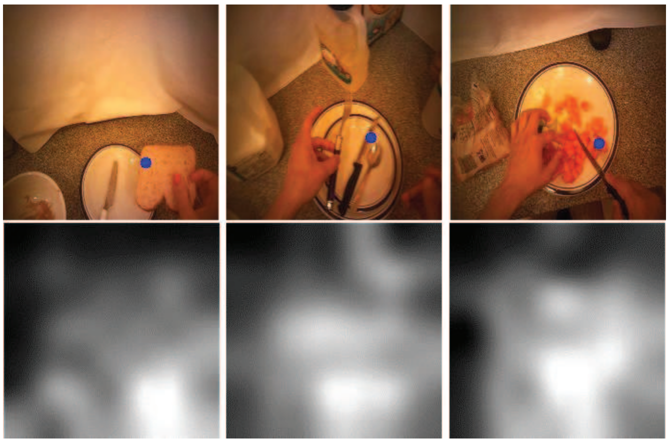


Fig. 6. First row: the video frames with GT gaze locations drawn in blue dots. The RGB-o model misclassifies the frames as ‘put knife’,‘take oil container’, and ‘cut mushroom’. The GT labels are ‘put bread’, ‘take knife’, and ‘cut tomato’. With the help of the predicted attention map (second row), our RGB-s model is able to recognize the actions correctly.

Figure 6 shows three examples that the RGB-o model misclassifies but the RGB-s model recognizes correctly. The first row is the video frames and we draw a blue dot in each frame to represent the ground truth (GT) gaze location. The second row is the visualized attention maps of the RGB-s model, which illustrates the region where it actually attends to. The values in each attention map are scaled so that the maximum value becomes 255 and the minimum becomes 0. The attention maps are then shown as black-white images and resized to the frame resolution. In these examples, the RGB-o model recognizes the objects that exist in the frames but are not being manipulated, and it considers them as the ‘noun’ part of the action. The attention map helps the RGB-s model to identify the object being manipulated and predict the correct action label. For example in the second column, the RGB-o model predicts the action to be ‘take oil container’ due to the appearance of the object in the upper part of the frame. The attention map enables the RGB-s model to focus on the lower part of the frame and obtain the correct label ‘take knife’.

E. Analysis of the Spatial Attention Network

Previous attention models do not have direct supervision on the predicted attention distribution [40], [41], [45]. The success of these attention models indicates that the models can implicitly learn an attention mechanism to focus on certain regions of the input to facilitate final prediction. Therefore, there are two natural questions. *Are these attention models helpful in egocentric action recognition tasks?* and *Does our spatial attention network learn better attention mechanism than these models with the help of human gaze?*

To answer these two questions, we design an attention network variant and test its performance on Gaze+ dataset. This model uses the same architecture as our spatial attention network to predict the attention distribution. However, we do not use the Euclidean loss with the ground truth attention map to train the convolution layer. Instead, we let the network learn the attention map prediction in the prediction-oriented manner by minimizing the final action recognition loss. We combine

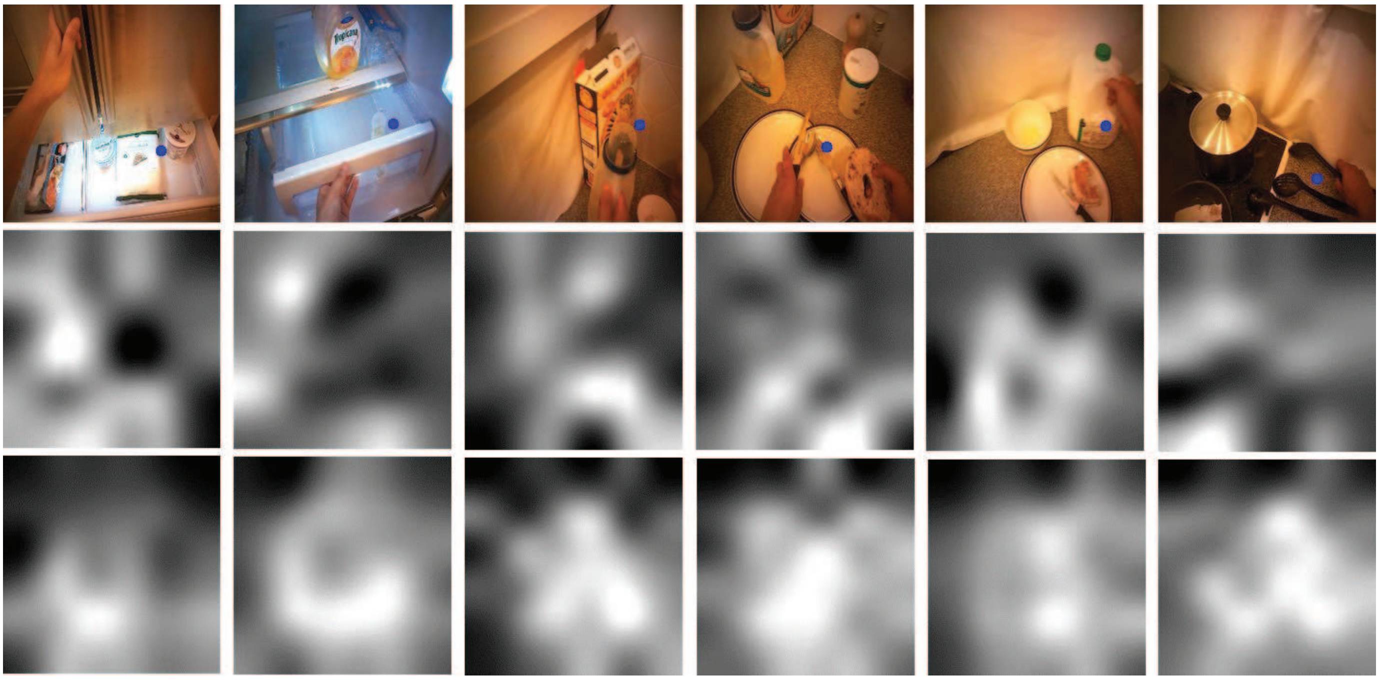


Fig. 7. The video frames with ground truth gaze locations drawn in blue dots (top), the visualized attention maps of RGB-v (middle), and the visualized attention maps of our spatial attention network RGB-s (bottom). The ground truth action label for each frame from left to right: open freezer, open fridge drawer, take oil container, take plate, take milk container, and take plastic spatula. Our model is able to attend to the image regions that are more relevant to the actions and is more consistent with the human attention (gaze).

TABLE III
COMPARISON OF OUR SPATIAL ATTENTION NETWORK (RGB-S) WITH THE VARIANT (RGB-V) THAT DOES NOT USE GAZE TO LEARN THE ATTENTION MECHANISM, AS WELL AS THE SALIENCY-BASED MODEL (RGB-SAL).

Methods	RGB-o	RGB-sal	RGB-v	RGB-s
Subject #1	49.85	52.47	50.41	51.65
Subject #2	72.26	66.79	70.80	72.63
Subject #3	51.22	49.89	50.78	51.00
Subject #4	62.00	57.14	60.29	62.29
Subject #5	57.64	57.64	61.11	61.81
Subject #6	52.74	54.85	56.12	55.27
Average	**56.39**	**55.41**	**56.65**	**57.42**

this attention prediction network with the generic CNN of the appearance stream, and we call this model RGB-v. The RGB-v model is similar to the attention networks used for spatial attention prediction in [40], [41], [45], which is a competitive baseline. Table III shows the action recognition performance of RGB-o, RGB-v and RGB-s methods on the Gaze+ dataset (the RGB-sal model is discussed later in Section IV-F). The RGB-v method outperforms the RGB-o model, which indicates that learning attention mechanism implicitly can facilitate egocentric action recognition to a certain extent. Our spatial attention network RGB-s is able to achieve higher accuracy than RGB-v. This demonstrates that using the human gaze to explicitly train the network can result in a better attention mechanism for egocentric action recognition.

We visualize the attention maps produced by the RGB-v and RGB-s methods, which are shown in Fig 7. The first row are the raw video frames with the ground truth human gaze locations drawn as blue dots. The second and third rows are the visualized attention maps of RGB-v and RGB-s methods. It can be found that our RGB-s model is able to attend to the image regions that are more relevant to the action and is more consistent with the human gaze and attention. For example, the first column represent the action of "open freezer". The RGB-v method tends to focus more on the upper region of the images, specially the hand regions. While our RGB-s method is able assign most of the high weights to the bottom region which represents the freezer.

F. Discussions

1) ***Comparison of the Spatial Attention Network with Saliency-Based Model:*** Our spatial attention network learns the top-down task-dependent human attention using gaze as supervision, which is able to focus on the task relevant regions for better action recognition. It would be interesting to see whether the models with the other type of attention [13], bottom-up attention, could help action recognition during these object manipulation tasks. Therefore, we compare our spatial attention network with the DeepSaliency model [52], which is well performing on several saliency detection datasets.

In the experiment, the saliency map produced by the DeepSaliency model is used to replace the attention map of our RGB-s model. The CNN feature map and the saliency map are kept fixed, while the fully-connected layer are re-trained for action prediction. We call this model "RGB-sal" and its performance on Gaze+ dataset is shown in Table III. The RGB-sal model produces lower average accuracy than both the RGB-v model and our RGB-s model. It is even worse than the RGB-o model which does not have any attention mechanism to weight its features. We give the following discussion and

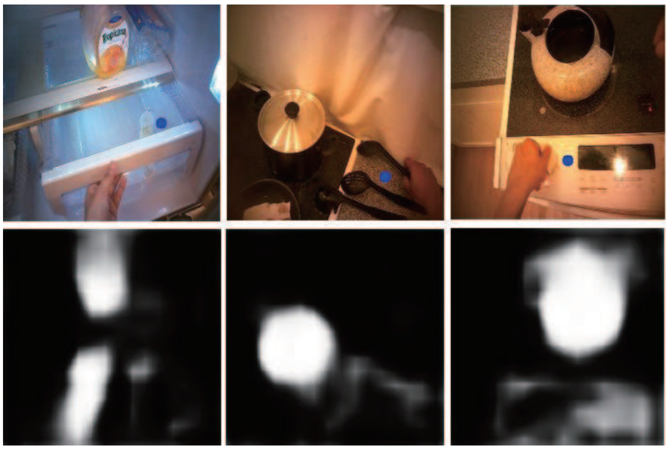


Fig. 8. The video frames with GT gaze location drawn in blue dots (top), and the saliency map generated by the DeepSaliency model (bottom). The GT labels from left to right: open fridge drawer, take plastic spatula, and turn on burner. The DeepSaliency model identifies the salient regions successfully, however, they are irrelevant to the current action. Assigning high weight to such regions could have adverse effect on action recognition.

explanation for this adverse effect of using saliency map to weight features.

As discussed in [12], [13], saliency belongs to the category of bottom-up human attention, which is mainly based on characteristics of a visual scene (stimulus-driven). It corresponds to the attention when humans are performing free-viewing on a scene or an image, in which the objects or regions that “stand out” relative to the neighboring parts attract human attention. The ground-truth of most saliency datasets is obtained by recording the locations where the human looks during free-viewing. In comparison, the human attention when they are performing certain tasks (e.g. the object manipulation tasks in the Gaze and Gaze+ dataset) belongs to the category of top-down attention, which is task-driven. The human eye movements and fixation during performing these tasks are very different from during free-viewing [11]. The points of eye fixation during such tasks are often not correspond to the most “salient” regions, but depends on the current task and human action [14].

Based on these discussions, we believe that the models used to predict saliency maps are not suitable for predicting attention maps in our experiments, where the human subjects are performing object manipulation tasks. This is because the two maps correspond to different types of human attention. The saliency models may focus on the salient but irrelevant regions of the frame, which could cause the adverse effect of worse performance than using features without any weight. Figure 8 shows some saliency maps that are inconsistent with task-dependent human attention. This experiment demonstrates the difference between the bottom-up and top-down attentions and provides interesting insights.

Temporal attention models have also been proposed for action recognition in third-person videos [43], [46], [42]. These models use different attention weights when aggregating the information of different frames for recognizing actions. The temporal weights can be computed based on the affinity scores of the features of different time steps [46] or learned using network layers [42]. Similar idea could be applied for egocentric action recognition. We tested the temporal attention models [46], [42] in our experiments, but obtained similar results as our original temporal network without the attention mechanism. We believe this is because the temporal structure of the egocentric actions in the current datasets is relatively simple and our bi-directional LSTM is enough for capturing such information. However, it is still an interesting future work to test these temporal attention models on future egocentric datasets.

TABLE IV
THE ARCHITECTURE STUDY OF OUR ATTENTION NETWORKS, WHICH COMPARES APPEARANCE STREAMS WITH ATTENTION NETWORK VARIANTS CONTAINING 1-6 CONVOLUTION LAYERS.

Methods	RGB-1L	RGB-2L	RGB-3L	RGB-4L	RGB-5L	RGB-6L
Subject #1	51.65	52.07	50.82	51.44	50.82	50.00
Subject #2	72.63	71.90	71.17	71.90	72.63	72.26
Subject #3	51.00	50.55	52.77	50.55	50.55	51.00
Subject #4	62.29	62.00	62.00	61.71	61.71	62.29
Subject #5	61.81	58.33	59.03	59.72	60.42	59.03
Subject #6	55.27	53.59	53.59	55.27	54.01	52.32
Average	**57.42**	**56.96**	**57.11**	**57.06**	**56.90**	**56.55**

TABLE V
THE ARCHITECTURE STUDY OF OUR ATTENTION NETWORKS, WHICH COMPARES MOTION STREAMS WITH ATTENTION NETWORK VARIANTS CONTAINING 1-6 CONVOLUTION LAYERS.

Methods	Flow-1L	Flow-2L	Flow-3L	Flow-4L	Flow-5L	Flow-6L
Subject #1	52.27	52.69	51.44	52.27	51.65	50.21
Subject #2	67.88	68.25	67.51	67.51	68.25	67.88
Subject #3	53.44	52.33	52.55	52.33	52.11	52.77
Subject #4	64.57	65.43	66.28	65.14	65.71	66.00
Subject #5	50.00	52.08	51.39	50.69	51.39	50.69
Subject #6	56.96	56.54	57.81	56.96	56.54	56.96
Average	**57.37**	**57.52**	**57.42**	**57.21**	**57.21**	**57.01**

*2) **Architecture Study of the Spatial Attention Network**:* It is non-trivial to determine how many convolution layers are needed for our spatial attention network to learn the human attention. Therefore we conduct this architecture study for our spatial attention network and test several variants with different numbers of convolution layers. Specifically, we combine the attention network variants containing 1-6 convolution layers with the generic CNNs, which results in the RGB-1L, RGB-2L, RGB-3L, RGB-4L, RGB-5L, RGB-6L, Flow-1L, Flow-2L, Flow-3L, Flow-4L, Flow-5L, and Flow-6L models. When there are 2 or more convolution layers in the model, the ReLU nonlinearity is used between the convolution layers. The results are shown in Table IV and Table V.

It can be seen that slightly better accuracies are achieved when using 2 or 3 layers for the flow-based stream. While there is a decreasing trend of the performance for both streams when there are more than 3 convolution layers in the attention network. Considering the results of both streams, the performances of the 1-layer models are better than others. The following two factors explain the reason for this result. First, more layers results in more parameters and increases the difficulty for network optimization. The currently available datasets might be too small for training and optimizing several convolution layers from scratch. We will still keep multi-layer attention network as an optional choice in the future, when sufficiently large dataset is available. Second, the feature map used in our spatial attention network is the output of the 13 convolution layers of VGG net, which is compact and discriminative. Based on the analysis about the correlation between human attention and action [11], [14], we believe this feature map is already relevant enough for both action recognition and attention prediction. Therefore, one layer is currently sufficient for predicting the attention map and we choose to use spatial attention network with one convolution layer in our experiments.

V. Conclusion

In this work, we propose a spatial attention network and a temporal network, which are incorporated in a two-stream architecture for egocentric action recognition. The temporal network utilizes bi-directional LSTM to model the long-range dependencies among the video frames. The spatial attention network learns to predict attention maps by using human gaze information as ground truth. The produced attention maps help our model to identify the most relevant parts in the frames and predict actions more accurately. By visualizing the attention maps of our spatial attention network and a comparing model, we demonstrate that our model is able to predict attention maps that are more consistent with human attention while performing the actions. Our model achieves competitive action recognition performance with the state-of-the-art methods on GTEA Gaze and GTEA Gaze+ datasets.

References

[1] Y. Wang, M. Lu, Z. Wu, L. Tian, K. Xu, X. Zheng, and G. Pan, "Visual cue-guided rat cyborg for automatic navigation [research frontier]," *IEEE Computational Intelligence Magazine*, vol. 10, no. 2, pp. 42–52, 2015.

[2] Z. Wu, G. Pan, and N. Zheng, "Cyborg intelligence," *IEEE Intelligent Systems*, vol. 28, no. 5, pp. 31–33, 2013.

[3] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 407–414.

[4] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2847–2854.

[5] A. Fathi, Y. Li, and J. M. Rehg, "Learning to recognize daily actions using gaze," in *European Conference on Computer Vision (ECCV)*, 2012, pp. 314–327.

[6] Y. Li, Z. Ye, and J. M. Rehg, "Delving into egocentric actions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 287–295.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[8] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] S. Singh, C. Arora, and C. Jawahar, "First person action recognition using deep learned descriptors," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] Y. Li, A. Fathi, and J. M. Rehg, "Learning to predict gaze in egocentric video," in *IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 3216–3223.

[11] A. L. Yarbus, "Eye movements during perception of complex objects," in *Eye Movements and Vision*, 1967, pp. 171–211.

[12] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 185–207, 2013.

[13] F. Katsuki and C. Constantinidis, "Bottom-up and top-down attention: different processes and overlapping neural systems," *The Neuroscientist*, vol. 20, no. 5, pp. 509–521, 2014.

[14] M. F. Land and M. Hayhoe, "In what ways do eye movements contribute to everyday activities?" *Vision research*, vol. 41, no. 25, pp. 3559–3565, 2001.

[15] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 16, 2011.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2005, pp. 886–893.

[17] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 428–441.

[18] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[19] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[20] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3169–3176.

[21] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1725–1732.

[22] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[23] L. Wang, W. Li, W. Li, and L. Van Gool, "Appearance-and-relation networks for video classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[24] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

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

[26] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2625–2634.

[27] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, "A hierarchical deep temporal model for group activity recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[28] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 20–36.

[29] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4489–4497.

[30] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4724–4733.

[31] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[32] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg, and V. Singh, "Gaze-enabled egocentric video summarization via constrained submodular maximization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2235–2244.

[33] J. Kopf, M. F. Cohen, and R. Szeliski, "First-person hyper-lapse videos," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, p. 78, 2014.

[34] X. Ren and C. Gu, "Figure-ground segmentation improves handled object recognition in egocentric video." in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, no. 3, 2010, p. 6.

[35] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, "Fast unsupervised ego-action learning for first-person sports videos," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3241–3248.

[36] A. Fathi and J. M. Rehg, "Modeling actions through state changes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2579–2586.

[37] Y. Zhou, B. Ni, R. Hong, X. Yang, and Q. Tian, "Cascaded interactional targeting network for egocentric video analysis," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1904–1913.

[38] J. Ba, V. Mnih, and K. Kavukcuoglu, "Multiple object recognition with visual attention," *arXiv preprint arXiv:1412.7755*, 2014.

[39] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4507–4515.

[40] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention." in *ICML*, vol. 14, 2015, pp. 77–81.

[41] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," *arXiv preprint arXiv:1511.04119*, 2015.

[42] W. Du, Y. Wang, and Y. Qiao, "Recurrent spatial-temporal attention network for action recognition in videos," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1347–1360, 2018.

[43] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot, "Skeleton-based human action recognition with global context-aware attention lstm networks," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1586–1599, 2018.

[44] H. Xu and K. Saenko, "Ask, attend and answer: Exploring question-guided spatial attention for visual question answering," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 451–466.

[45] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 21–29.

[46] Y. Peng, Y. Zhao, and J. Zhang, "Two-stream collaborative learning with spatial-temporal attention for video classification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[47] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 2017–2025.

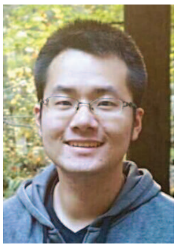
[48] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

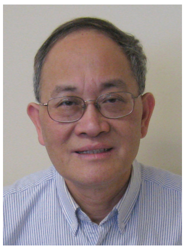
[50] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[51] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 3551–3558.

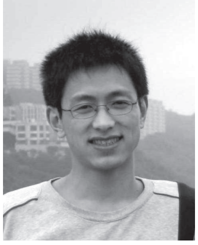
[52] X. Li, L. Zhao, L. Wei, M.-H. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang, "Deepsaliency: Multi-task deep neural network model for salient object detection," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3919–3930, 2016.



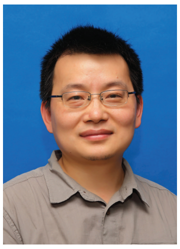
Minlong Lu received the Ph.D. degree in computer science from both Simon Fraser University, Canada and Zhejiang University, China, in 2018. He received the B.Eng. degree from Zhejiang University in 2011. He was with the Vision and Media Lab at Simon Fraser University during his graduate study and is currently a postdoctoral research assistant in the same lab, both advised by Prof. Ze-Nian Li. His research interests include computer vision and machine learning.



Ze-Nian Li is a Professor in the School of Computing Science at Simon Fraser University, British Columbia, Canada. Dr. Li received his undergraduate education in Electrical Engineering from the University of Science and Technology of China, and M.Sc. and Ph.D. degrees in Computer Sciences from the University of Wisconsin-Madison under the supervision of the late Professor Leonard Uhr. His current research interests include computer vision, multimedia, pattern recognition, and artificial intelligence. He is the co-Director of the Vision and Media Lab at Simon Fraser University. He is also the co-author of the book "Fundamentals of Multimedia", 2nd ed., published by Springer, 2014.



Yueming Wang received the Ph.D. degree from Zhejiang University, Hangzhou,China, in 2007. He was a Post-Doctoral Fellow with the Department of Information Engineering, Chinese University of Hong Kong, Hong Kong, from 2007 to 2010. He was an associate professor from 2010 to 2016 in the Qiushi Academy for Advanced Studies (QAAS), Zhejiang University, P.R. China. He is now a professor in QAAS. His research interests are brain-machine interface, data mining, and pattern recognition.



Gang Pan (M'05) received the B.Eng. and Ph.D. degrees from Zhejiang University, China, in 1998 and 2004, respectively. He is currently a professor of the Department of Computer Science, and deputy director of State Key Lab of CAD&CG, Zhejiang University, China. From 2007 to 2008, he was a visiting scholar at the University of California, Los Angeles. His current interests include artificial intelligence, brain-inspired computing, brain-machine interfaces, pervasive computing, and computer vision. He has authored over 100 refereed papers, and 35 patents granted. Dr. Pan received three best paper awards and three nominations from premier international conferences. He is the recipient of IEEE TCSC Award for Excellence (Middle Career Researcher), CCF-IEEE CS Young Computer Scientist Award, and National Science and Technology Progress Award. He serves as an Associate Editor of IEEE Trans. Neural Networks and Learning Systems, IEEE Systems Journal, Pervasive and Mobile Computing, and ACM Proceedings of Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT).