Learning Spatiotemporal Attention for Egocentric Action Recognition

Minlong Lu[1,2] Danping Liao[3] Ze-Nian Li[1]
[1]School of Computing Science, Simon Fraser University, Canada
[2]Huawei Technologies, Canada
[3]College of Computer Science and Technology, Zhejiang University, China

Abstract

Recognizing camera wearers' actions from videos captured by the head-mounted camera is a challenging task. Previous methods often utilize attention models to characterize the relevant spatial regions to facilitate egocentric action recognition. Inspired by the recent advances of spatiotemporal feature learning using 3D convolutions, we propose a simple yet efficient module for learning spatiotemporal attention in egocentric videos with human gaze as supervision. Our model employs a two-stream architecture which consists of an appearance-based stream and motion-based stream. Each stream has the spatiotemporal attention module (STAM) to produce an attention map, which helps our model to focus on the relevant spatiotemporal regions of the video for action recognition. The experimental results demonstrate that our model is able to outperform the state-of-the-art methods by a large margin on the standard EGTEA Gaze+ dataset and produce attention maps that are consistent with human gaze.

1. Introduction

With the increasing popularity of wearable cameras, there is a growing interest in recognizing actions using the first-person/egocentric videos, which has potential applications including remote assistance, health monitoring and human-robot interaction. The wearable camera is usually mounted on the person's head with its optical axis aligned with the wearer's field of view. Action recognition for the camera wearer using the first-person videos is different from that in third-person setting. First, unlike in the third-person video, the camera wearer's pose are mostly unavailable in egocentric videos. The recognition of egocentric actions often requires more fine-grained discrimination of the objects being manipulated and their locations. Second, strong egomotions are often present in egocentric videos due to the head motion of the person, whereas the third-person videos are usually static or more stable. These aspects make action recognition in egocentric videos very challenging.

Figure 1. Sample frames and gaze locations from EGTEA Gaze+ dataset. The gaze locations are drawn as blue dots. It is sufficient to recognize the actions (cut lettuce, operate stove), by only looking at the region around the gaze point within the blue circle.

Egocentric actions are usually defined as verb-object pairs (e.g. take bread) and recognized from trimmed videos. Traditional models explore a variety of hand-crafted features for egocentric action recognition, such as object-centric features [21] and egocentric cues [15], which are shown to be complementary. Recent works incorporate advances in deep neural networks for egocentric action recognition [17, 23, 16, 14]. Information such as hand masks [23] and localized objects [17] are employed in these models to facilitate action recognition.

Attention mechanism is also utilized to guide the networks to focus on the relevant regions for egocentric action recognition [16, 14, 24, 25]. These models usually predict an attention distribution, based on which they either re-weight the features or select the features with highest attention. Some of these attention models are trained in a goal-oriented manner, by attempting to minimize the final prediction error of the task [24, 25]. Therefore these models implicitly learn an attention mechanism in favor of the final prediction. Other attention models are trained using human gaze as direct supervision. The motivation is that eye movements reflect a person's thinking process and represent human attention [37]. It is demonstrated in [13] that during object manipulation tasks a substantial percentage of fixations fall on the task-relevant regions. Examples are

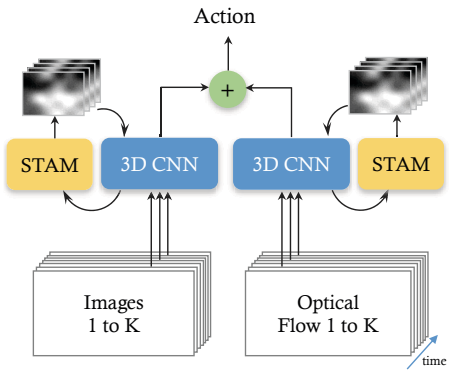

Figure 2. Our model employs a two-stream architecture. Each stream has a spatiotemporal attention module (STAM) for attention prediction, which helps the model identify relevant spatiotemporal regions for action recognition.

shown in Figure 1, where only a small region around the gaze point is sufficient for recognizing the action. Focusing on these regions may even reduce the potential misleading from the clutter background. Therefore, training attention models with gaze supervision enables the learning of the task-dependent top-down attention [3, 11], and results in better attention predictions [16, 14].

Based on these insights, we propose a spatiotemporal attention module (STAM) which contains inception blocks [26] with 3D convolutions to learn spatiotemporal attention with human gaze as supervision. Our STAM is incorporated in a two-stream architecture for egocentric action recognition. Inspired by the recent advances of spatiotemporal feature learning in videos using 3D CNNs [28, 4], we adopt the I3D model [4] as our two-stream backbone network. Each stream of I3D is augmented with a spatiotemporal attention module (STAM) for attention prediction, which helps the model to identify relevant spatiotemporal regions for action recognition. Our model shares the similar idea as the model proposed in [16], which uses gaze supervision to train a spatial attention network and achieve good performance. The shortcoming of the method in [16] is that its attention prediction is restricted in the spatial domain, where the attention map is generated by convolutional layers solely based on the current frame. We believe that both the video data and the human gaze are usually temporally consistent. It would be beneficial to consider the information in the nearby frames when predicting attention for the current frame. Our model can be considered as a generalization of [16] by extending both the feature learning and attention prediction to spatiotemporal domain. The overview of our model is shown in Figure 2.

Our contribution can be summarized as follows: (1) We propose a spatiotemporal attention module (STAM), which is incorporated in a two-stream model for egocentric action recognition. This model is able to outperform the state-of-the-art methods by a large margin on the EGTEA Gaze+ dataset. (2) We provide detailed ablation analysis to demonstrate how the proposed spatiotemporal attention module contributes to the performance. (3) We compare our STAM to a goal-oriented attention model and demonstrate both quantitatively and qualitatively that our model is capable of learning better attention mechanism for egocentric action recognition.

2. Related Work

2.1. Action Recognition and Egocentric Vision

Action recognition has been one of the key problems in computer vision. The majority of research focuses on recognizing human actions in third-person videos. A large number of features have been designed for action recognition, such as histogram of oriented gradients (HOG) [6] and motion boundary histograms (MBH) [30]. Deep neural networks are also widely used for action recognition. The idea of two-stream architecture is proposed in [22], which feeds RGB frames and optical flow images into separate CNN streams and fuse the scores to recognize actions. Recurrent networks are used on top of the CNNs for modeling temporal dependencies for action recognition [8, 31]. The convolution and pooling operations of CNNs are extended to 3D in models such as C3D [28] and I3D [4], which enables the spatiotemporal feature learning from video inputs. The 3D operations can be factorized into separate spatial and temporal components to facilitate optimization [29, 34].

With the advent of various wearable cameras, the research on egocentric vision topics has attracted a lot attention, such as video summarization [35], object recognition [27], as well as action recognition [12, 21]. Researchers have designed object-centric features [21] and egocentric cues [15], and also use motion compensation [15] to recognize egocentric actions. Recent works have attempted to employ CNNs to tackle this problem [17, 23, 9]. Stacked input of hand mask, homography image, and saliency maps are used as input of the Ego convnet model [23]. In [17], networks are trained to segment hand and localize object, and then use the information to facilitate the recognition of actions, which have "verb+object" form. The information used in these models often requires additional annotations which are expensive to obtain. Using eye-tracking devices, the gaze or eye fixation of the person can be recorded during object manipulation tasks, which is relatively easier to acquire. Human gaze is often utilized together with the attention mechanism and is shown to be helpful in egocentric action recognition [9, 14, 16].

2.2. Visual Attention Model

Attention models have been used in tasks such as machine translation [2], speech recognition [5], and are also

proved successful in variant vision tasks [32], such as object recognition [1], image captioning [36], as well as action recognition [39]. The visual attention models aim at identifying relevant spatial regions in the visual input and highlight these regions to facilitate the task, which mimics the human perception and thinking process. These models usually predict a probability distribution over a grid of features, which represents the level of attention on each region. Then the attention distribution is used to either re-weight the features or select the features with highest attention [36]. The spatial transformer networks [10] allows to attend to arbitrary regions of the data by introducing affine transformations to the feature maps. Attention is applied both spatially and channel-wise in [33]. Attention mechanism can be generalized to the temporal domain, where the models learn to assign different weights to the video segments for action recognition [20].

Similar ideas has been employed and extended in egocentric action recognition [25, 24, 9, 14, 16]. A spatial attention is learned for each frame using class activation maps in [25]. The Long short-term attention model [24] further incorporated attention into convLSTM to track the attention temporally. Like other attention models in third-person action recognition, these models are trained in a goal-oriented manner by minimizing the final prediction error. Therefore the attention mechanism is learned implicitly to favor action recognition.

It has been shown that gaze represents human attention and is highly coordinated with actions [37, 13]. Gaze information has been utilized to guide the training of attention prediction and facilitate egocentric action recognition [16, 14, 9]. A spatial attention network is proposed in [16], which is incorporated in a two-stream network to produce attention maps. Gaze behavior and actions can be jointly modeled [14, 9], where the two tasks benefit each other because of their underlying correlation. Gaze is described as probabilistic variable to model the uncertainty in [14]. The attention map in [9] is produced by several convolutional kernels based on predicted actions. Training attention models with gaze supervision enables the learning of the task-dependent top-down attention [3, 11], and can produce better action recognition performance [16, 14]. Our model shares similar idea as [16] while we extend both the feature learning and attention prediction to spatiotemporal domain and achieve significant performance boost.

3. The Proposed Method

In this work, we propose a spatiotemporal attention module (STAM), which employs inception blocks [26] with 3D convolutions to learn the spatiotemporal attention directly from the feature volume with gaze as ground truth. The attention map is used to help our model to selectively focus on the relevant part of the data to recognize actions. The STAM

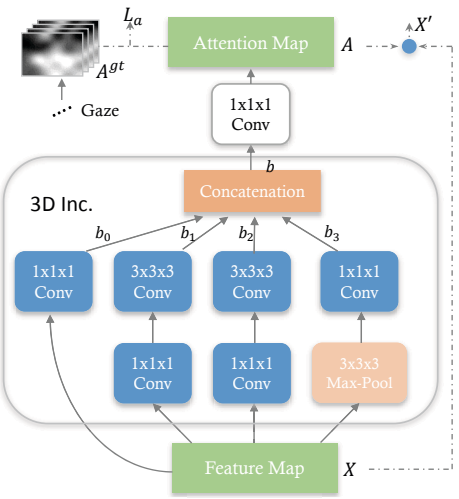

Figure 3. Our spatiotemporal attention module (STAM) employs a 3D inception module (3D inc.) and a 3D convolution layer to predict attention map.

is incorporated in a two-stream architecture, which can be considered as a generalization of [16] by extending both the feature learning and attention prediction to spatiotemporal domain. In this section, we will describe our spatiotemporal attention module and provide detailed framework of our two-stream architecture.

3.1. Spatiotemporal Attention Module

Our spatiotemporal attention module consists of a 3D inception module [26, 4] and a 3D convolution layer, as shown in Figure 3. It takes the feature map $X \in \mathbf{R}^{C\times T\times H\times W}$ as input and output an attention map $A \in \mathbf{R}^{T\times H\times W}$ as follows:

$$\begin{aligned} b_0 &= conv1\_0(X) \\ b_1 &= conv3\_1(conv1\_1(X)) \\ b_2 &= conv3\_2(conv1\_2(X)) \\ b_3 &= conv1\_3(max\_pool(X)) \\ b &= concat([b_0; b_1; b_2; b_3]) \\ A &= f(conv1\_a(b)), \end{aligned} \tag{1}$$

where C denotes number of input channels and T, H, W is the resolution of the spatiotemporal feature volume, $conv3\_i$ denotes 3D convolution with $3 \times 3 \times 3$ kernel, $conv1\_i$ denotes 3D convolution with $1 \times 1 \times 1$ kernel, and f represents a linear function that scales the input to $[0, 1]$. The 3D inception module has 4 convolutional branches, which take X as input and produce intermediate feature maps $b_i \in \mathbf{R}^{C_i\times T\times H\times W}, i = 0, 1, 2, 3$. These intermediate feature maps are concatenated channel-wise to produce $b \in \mathbf{R}^{C_b\times T\times H\times W}$, with $C_b = \sum_{i=0}^{3} C_i$. The feature b is then process by $conv1\_a$ and the scale function f to produce the spatiotemporal attention map A. The feature map A is

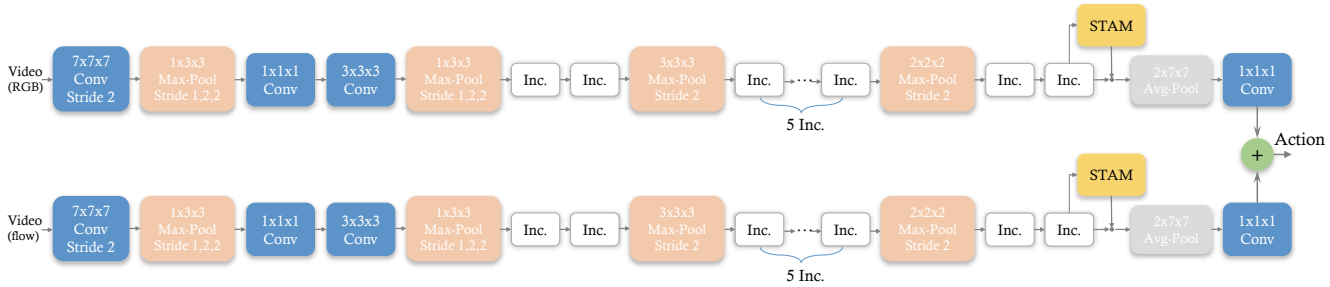

Figure 4. Framework details of our two-stream model. Each stream takes RGB/flow video as input and incorporates our spatiotemporal attention module (STAM) to prediction attention map. Inc. represents 3D inception module which is shown in Figure 3.

a 3D volume with resolution T, H, W, which indicates a spatial attention $A_t \in \mathbf{R}^{H\times W}$ for each time stamp t. Unlike previous spatial attention models [16, 25] which generate A_t solely based on the frame I_t, our model takes the advantage of 3D convolutions and uses spatiotemporal features of consecutive frames to predict A simultaneously. This simple idea models both spatial and temporal information for attention prediction, without the need of recurrent tracking of the attention maps in [24].

In this work, we utilize the human gaze information as ground-truth to guide the training of our spatiotemporal attention module (STAM). This is motivated by the coordination of gaze behavior and human action [13] and shown to be effective in previous attention models [16, 14]. The gaze or eye fixation can be obtained by first tracking the eye movement using wearable tracking device (e.g. Tobii eye tracking glasses), and then synchronize and project it onto the image plane, which is represented as the gaze location (x, y) in each frame. We generate the ground-truth attention map $A^{gt} \in \mathbf{R}^{T\times H\times W}$ by applying a Gaussian bump around the gaze points. This simple process is able to handle the uncertainty of gaze to certain extent and facilitate attention training. The attention weight a^{gt}_{ijt} of A^{gt} at time t and spatial location i, j is computed by:

$$a^{gt}_{ijt} = e^{-\frac{(i-x'_t)^2+(j-y'_t)^2}{2\sigma^2}}, \tag{2}$$

where x'_t, y'_t are the scaled gaze coordinates in the spatial resolution at time t and σ is set to be $H/2$. We use Mean Square Error to measure the difference between the predicted attention A and ground-truth attention A^{gt}, which is used during back-propagation to train the attention learning. Note that the ground-truth attention is only used in the training phase. The MSE loss is computed and averaged over all a_{ijt} in A as:

$$L_a = \frac{1}{THW}\sum_{t=1}^{T}\sum_{i=1}^{H}\sum_{j=1}^{W}(a_{ijt} - a^{gt}_{ijt})^2. \tag{3}$$

We combine the predicted attention map A with the feature map X to produce a more informative feature map X' as follows:

$$X'_c = A \odot X_c. \tag{4}$$

where $X'_c, X_c \in \mathbf{R}^{T\times H\times W}, c = 1...C$ is one channel of the feature map, $\odot$ represents element-wise multiplication of corresponding entries a_{ijt} and $X_{c,ijt}$. In this attention process, the more relevant spatiotemporal feature in X' is assigned higher weights, which is then processed by following layers to recognize actions.

3.2. The Two-Stream Architecture

The two-stream idea is first proposed in [22] and has been widely used in action recognition models for both third-person and first-person videos [16, 8, 4, 29]. The two streams learn appearance and motion features from RGB and flow inputs separately, which are shown to be complementary for action recognition. Even when each stream uses 3D convolution network with spatiotemporal feature learning ability, the two stream idea is still beneficial [4, 29].

In this work, we employ the two-stream I3D model [4] as our backbone network, which extend Inception-V1 [26] to be 3D convolution network. Our two-stream model contains a RGB stream and a flow stream, which takes RGB and flow video segments as input. Each stream incorporates our STAM to predict a spatiotemporal attention map, which is then used to weighted average pool the feature map. The scores of the two streams are fused for the final action prediction, as shown in Figure 4.

4. Experiments

4.1. Dataset and Experimental Setup

Dataset. We evaluate our proposed model on the Extended GTEA Gaze+ dataset (EGTEA Gaze+) [14], which is currently the standard and largest egocentric dataset that contains gaze data. This dataset is collected with a head-mounted camera and the actions involve hand-object interactions. Each action is represented by a verb and a set of nouns, for example "put bread (on) container". The human gaze is tracked using SMI eye tracking glasses and projected to an image coordinate in each frame, indicating the loca-

tion where the person is looking at. EGTEA Gaze+ contains 28 hours of videos with frame rate 24fps, which is from 86 unique sessions performed by 32 subjects. There are 106 action categories and a total of 10321 action instances. Each action instance is a trimmed video segment during which the camera wearer completes one action. Each instance and all the frames in it have a single action label.

Some previous methods use GTEA Gaze and GTEA Gaze+ datasets, which are not used in this work. The reason is that GTEA Gaze+ is a subset of EGTEA Gaze+ dataset, and GTEA Gaze dataset is too small (total 331 instances) and suffers from limited and imbalanced data problem as discussed in [15, 16]. The largest dataset in egocentric vision is the EPIC-Kitchens dataset [7], which contains 55 hours of videos and 39594 action instances. This dataset does not have gaze information, therefore we are not able to evaluate our model on this dataset.

Evaluation Metric. The egocentric action recognition is tackled as a classification problem for trimmed video segments. There are two metrics used in the previous methods to evaluate egocentric action recognition performance: micro-average accuracy and macro-average accuracy, which are sometimes referred to as *accuracy* and *mean class accuracy* in the literatures, respectively. We would like to give a clarification in case there is any confusion. These two metrics are defined as follows:

$$Micro = \frac{\sum_{i=1}^{K} c_i}{\sum_{i=1}^{K} n_i}, \quad Macro = \frac{1}{K}\sum_{i=1}^{K}\frac{c_i}{n_i}, \qquad (5)$$

where K is the total number of classes, c_i denotes the number of instances of class i that are correctly recognized, n_i denotes the total instance number of class i. The micro and macro accuracy will be equal when all classes have the same number of instances n_i. While for imbalanced data, they can be different. For example, a 2-class classification problem with 98% samples belonging to class 1. Simply predicting all samples with label 1 will have an micro accuracy of 98%. In comparison, the macro accuracy is 50%. The macro accuracy or mean class accuracy is similar to the idea of confusion matrix, which provides additional insight about the action recognition performance. In our experiments, we use micro accuracy, macro accuracy as well as confusion matrix to evaluate our model.

Implementation Details. Our model is implemented using Pytorch [19] based on the I3D model [4]. All the video frames are resized to the resolution of 320×240. We compute optical flow using TV-L1 algorithm [38], and we truncate the flow values to the range of [-20, 20] and scale them to [-1, 1]. The pixel values of the RGB frames are also scaled to [-1, 1]. The optical flow images have 2 channels. The first channel is x-component of the flow vector and the second channel is the y-component. During training, we perform data augmentation by randomly cropping 224×224 patches of the input video clips and randomly flipping horizontally. The gaze location are refined according to the augmentation performed to the frames. During test, we perform center crop to the videos and do not flip the data.

The input video length to each stream is chosen to be 16 frames, which is the number of frames of the shortest action instance. For optical flow videos of such instances, which has only 15 frames, we repeat the last flow image. At training time, we randomly select a start frame for an instance and use the consecutive 16 frames as a training sample. At test time, we extract 16 frame clips with a stride of 8 frames from each testing instance and compute the frame scores, and the scores of the overlapped 8 frames are averaged. After evaluating all the 16 frame clips of an instance, the scores of all the frames are averaged for predicting the instance label.

Similar to the training procedure in [4], we train each stream using stochastic gradient descent with momentum set to 0.9 and weight decay set to 10^{-7}. We use batch size 12 and an initial learning rate of 0.1. The learning rate is decreased by a factor of 0.1 after 1.2k and 4k iterations, and the model is trained for 64k iterations. The dropout rate is set to 0.5 and is performed after average pool and before the final prediction layer, which is a 3D convolution with kernel size 1 as [4]. The models are first pretrained on imagenet and kinetics dataset [4], and then trained on the target EGTEA Gaze+ dataset. Using this training scheme, we are able to produce better egocentric action recognition performance of I3D than some previous reproductions. Base on this backbone model, we add our spatiotemporal attention module (STAM) and use the same training scheme with the loss weight of the attention prediction set to 0.1. Please refer to our code for all the details. (`https://github.com/ymlml/STAM`)

4.2. Comparison with Previous Methods

We evaluate our model on the split1 of EGTEA Gaze+ dataset (8299 training and 2022 testing instances), which follows previous methods [14, 9]. We report both micro and macro average accuracy of our methods, as shown in Table 1. It can be seen that our two-stream fusion model achieves a large improvement over the individual RGB and flow stream. This demonstrates that the two-stream idea is still effective with 3D CNNs and can learn complementary information for action recognition. The spatial attention network (SAN) model proposed in [16] uses VGG net as the backbone network. In our experiments we implement this model using Inception V1 [26], which is better performing than VGG net. Therefore, this model is the reduced version of our model by learning feature and attention only spatially. Our model achieves significantly better results than the SAN model, which demonstrates the effectiveness of spatiotemporal feature learning and attention modeling.

Table 1. Comparison of our method with previous methods on the split1 of EGTEA Gaze+ dataset.

Methods	Micro (%)	Macro (%)
MCN [9]	55.63	-
SAN-RGB [16]	52.91	42.72
SAN [16]	57.10	46.84
TSN [31]	58.01	-
Ego-RNN [25]	62.17	-
Li *et al.* [14]	-	53.30
EleAttG [39]	57.01	-
LSTA-RGB [24]	57.94	-
LSTA [24]	61.86	-
Ours: RGB stream	63.56	56.34
Ours: flow stream	60.09	50.99
Ours: two-stream	**68.60**	**60.54**

'-' denotes that the model did not provide result in this metric.

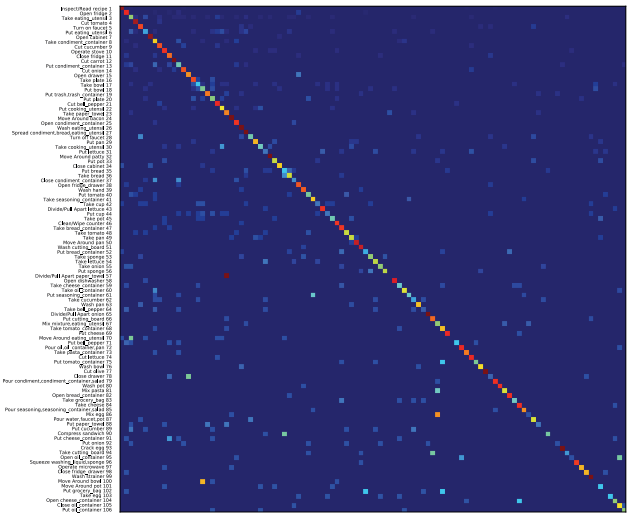

Figure 5. Confusion matrix of our model for all 106 action classes on EGTEA Gaze+ dataset. Please see Figure 6 for the color bar.

We also compare our method with other well-performing methods, as shown in Table 1. The TSN [31] was proposed for action recognition from third-person videos and adapted for egocentric recognition. The EleAttG [39] is a generic method for employing attention mechanism into RNN models. The MCN [9], Li *et al.* [14], Ego-RNN [25], and LSTA [24] were proposed for egocentric action recognition. The accuracy of TSN is from [25] and the accuracy of EleAttG is from [24]. The results of MCN, Li *et al.*, Ego-RNN, and LSTA are from their original papers. Our model is able to outperform all other methods by a large margin and achieves state-of-the-art results in terms of both micro and macro accuracy. The confusion matrices of our two-stream model are shown in Figure 5 and Figure 6. The action categories are sorted based on decreasing number of instances. Figure 5 includes the results of all 106 action categories of EGTEA Gaze+ dataset, while Figure 6 is the

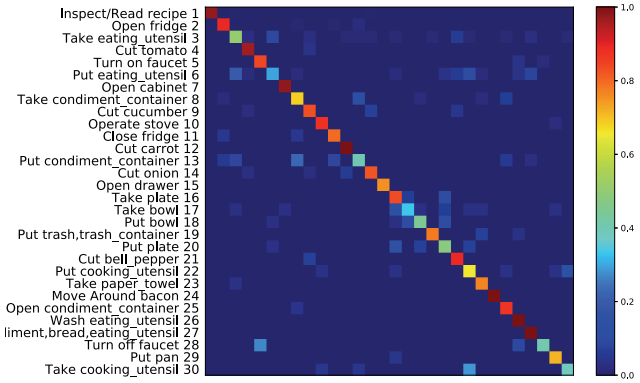

Figure 6. The enlarged confusion matrix of Figure 5, which shows the results for the first 30 action categories.

enlarged version showing the first 30 actions. It is demonstrated that our method is able to get most of the categories correctly recognized.

Figure 7 demonstrates sample frames from 9 action instances from the testing set and their corresponding attention maps produced by our model. We choose to show two frames for each instance and draw the ground truth gaze locations as a blue dot in each frame. We visualize the attention map by first scaling the values to the range of [0, 255]. Then the attention maps are resized to the same resolution as the frames and shown as black-white images. The attention maps illustrate the regions where our model actually focuses on. It can be found that these regions are relevant to the current actions and are consistent with human gaze and attention. Take the last action instance in Figure 7 as an example, which has ground truth action label "take bowl". Although there seems to be a more salient object (towel with flowers patterns) on the right of the frames, our model is able to produce high attention weights around the object being manipulated (bowl) and recognize the action correctly. This demonstrates that with the help of gaze/human attention during training, our model learns the task-dependent action and predict correct attention map during testing time.

4.3. Ablation Study

To analyze the performance of the spatiotemporal attention module (STAM) and how it contributes to the recognition accuracies in our two-stream model, we conduct a detailed ablation study by testing the performance of each stream with and without the STAM. The ablation study is conducted on all the 3 splits of the EGTEA Gaze+ dataset. The models evaluated in this study are listed below:

1. **RGB-o**: this model contains only 3D CNN inception V1, which is the RGB stream of I3D.

2. **RGB-a**: this model is our RGB stream which contains the 3D CNN and our STAM.

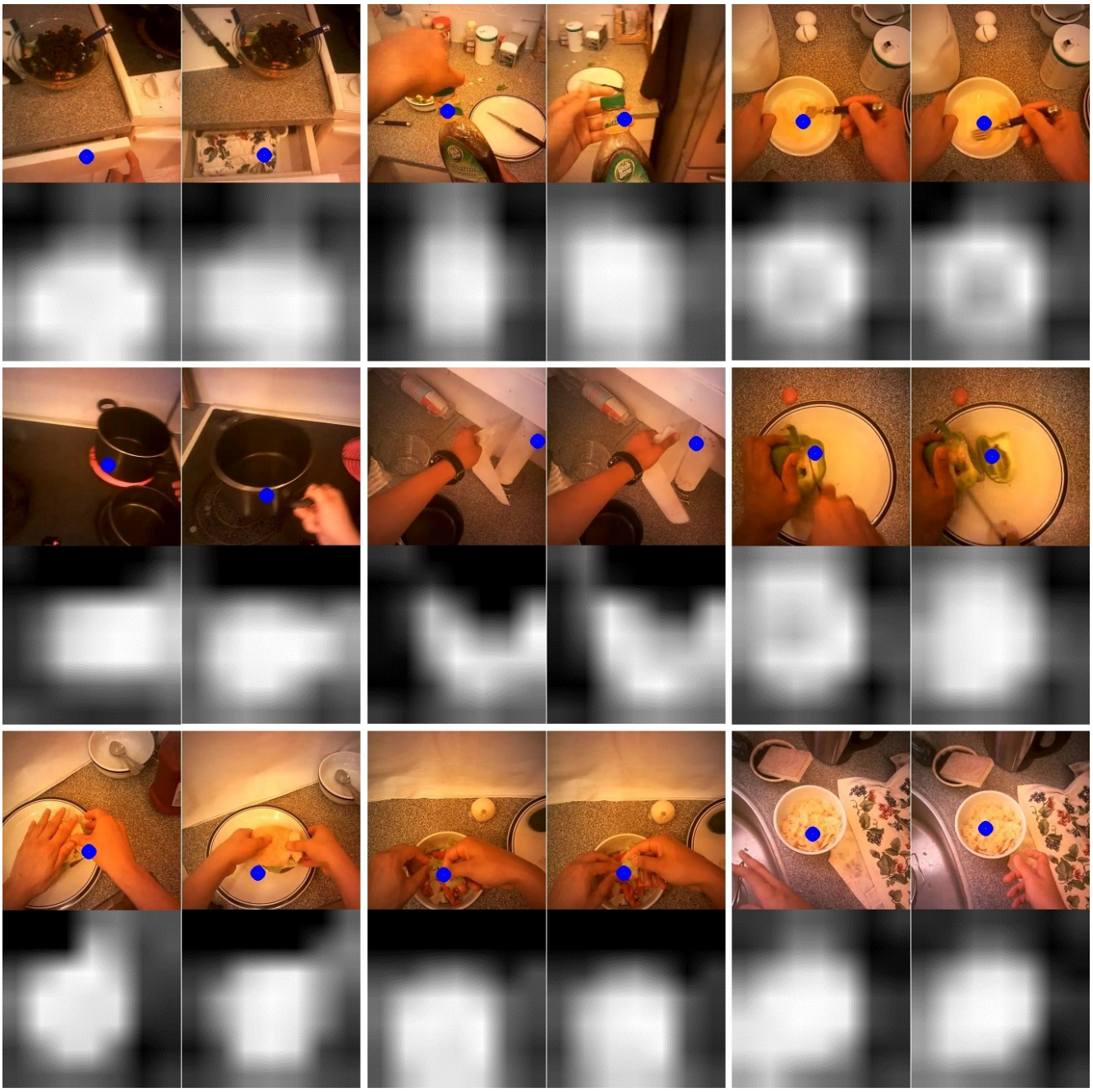

Figure 7. The sample frames and corresponding visualized attention maps from 9 action instances from the testing set of EGTEA Gaze+ dataset. The ground truth gaze locations are drawn as a blue dot in each frame. The ground truth action labels for each instance from left to right are: open drawer, close condiment container, mix egg (top), move around pot, pull apart paper towel, cut pepper (middle), compress sandwich, pull apart lettuce, take bowl (bottom). Our model is able to attend to the image regions that are relevant to the actions and is consistent with the human attention (gaze).

3. **Flow-o**: this model contains only 3D CNN inception V1, which is the flow stream of I3D.
4. **Flow-a**: this model is our flow stream which contains the 3D CNN and our STAM.
5. **Fuse-o**: the score fusion from RGB-o and Flow-o, which is the two-stream I3D.
6. **Fuse-a**: the score fusion from RGB-a and Flow-a, which is our two-stream model.

The detailed results of these models are listed in Table 2. We can see that the "-a" models are able to outperform the

Table 2. Detailed ablation study of our method on EGTEA Gaze+ dataset. There are 3 splits of this dataset and we produce micro and macro accuracy on each split and compute the average. The numbers are in percentage.

Methods	RGB-o	RGB-a	Flow-o	Flow-a	Fuse-o	Fuse-a
Micro: Split 1	62.51	63.65	57.86	60.09	67.56	68.60
Micro: Split 2	59.69	61.08	54.15	55.79	64.09	65.33
Micro: Split 3	58.39	59.03	53.19	55.02	63.09	63.98
Micro: Average	60.20	61.25	55.07	56.97	64.91	65.97
Macro: Split 1	54.89	56.34	46.96	50.99	59.26	60.54
Macro: Split 2	50.48	51.28	42.05	44.67	53.74	55.21
Macro: Split 3	50.02	50.81	42.08	45.07	53.36	55.32
Macro: Average	51.80	52.81	43.70	46.91	55.45	57.02

Table 3. The performance of the models without using gaze supervision for STAM.

Methods	RGB-v	Flow-v	Fuse-v
Micro (%): Split 1	62.66	59.30	67.70
Macro (%): Split 1	55.30	49.70	59.73

"-o" models, which demonstrates that our STAM is able to use the attention mechanism to facilitate action recognition. The best performance is achieved by the Fuse-a model, which fuses the scores of our RGB stream and flow stream.

4.4. Analysis of the Gaze Supervision

Our spatiotemporal attention module is trained using ground truth attention map generated using human gaze. In order to analyze the beneficial of the gaze supervision, we design a variant of our model by removing the gaze supervision during training. Therefore, this model learns the attention prediction implicitly in the goal-oriented manner by minimizing the final action recognition loss. We use RGB-v, Flow-v, and Fuse-v to represent the RGB stream, flow stream and the two-stream model without gaze supervision. The performance of these models are shown in Table 3.

The "-v" models achieve slightly better results than the corresponding "-o" models models, which indicates that learning attention mechanism implicitly can facilitate egocentric action recognition. The performance of our "-a" models is better than the "-v" models, which demonstrates that using the human gaze to explicitly train the network can result in a better attention mechanism. This is also verified from previous works [14, 16].

Besides the quantitative evaluation, we attempt to qualitatively evaluate the "-v" and "-a" models by visualizing the feature vectors of the average pooling layer, which has a dimension of 1024. We extract the feature vectors of the testing instances using the Flow-v and Flow-a models, which are then projected to 2-dimensional space using t-SNE [18]. We only show the feature vectors of the top 10 frequent action classes in order to make the figure more compact. The separability of these feature vectors in their respective spaces seems similar for the "-a" and "-v" models. This is justifiable since the improvement of the "-a" model over the "-v" model is not large enough to show visually.

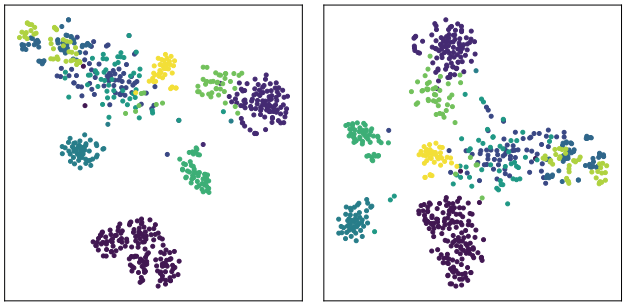

Figure 8. The visualized feature vectors of Flow-v (left) and Flow-a (right) models using t-SNE [18]. Each action instance is visualized as a point and instances belonging to the same class have the same color.

5. Conclusion

In this work, we propose a spatiotemporal attention module (STAM), which is incorporated in a two-stream model for egocentric action recognition. Our STAM learns to predict spatiotemporal attention by using human gaze as ground truth. The STAM is able to identify the relevant regions and help our model to recognize actions more accurately. The visualized results demonstrate that the attention maps are consistent with human gaze and are good for action recognition. Our model outperforms the state-of-the-art methods by a large margin on the standard EGTEA Gaze+ dataset.

References

[1] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[3] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):185–207, 2013.

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, 2017.

[5] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Advances in Neural Information Processing Systems*, pages 577–585, 2015.

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The epic-kitchens dataset. In *European Conference on Computer Vision (ECCV)*, 2018.

[8] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2625–2634, 2015.

[9] Yifei Huang, Minjie Cai, Zhenqiang Li, and Yoichi Sato. Mutual context network for jointly estimating egocentric gaze and actions. *arXiv preprint arXiv:1901.01874*, 2019.

[10] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

[11] Fumi Katsuki and Christos Constantinidis. Bottom-up and top-down attention: different processes and overlapping neural systems. *The Neuroscientist*, 20(5):509–521, 2014.

[12] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3241–3248, 2011.

[13] Michael F Land and Mary Hayhoe. In what ways do eye movements contribute to everyday activities? *Vision research*, 41(25):3559–3565, 2001.

[14] Yin Li, Miao Liu, and James M Rehg. In the eye of beholder: Joint learning of gaze and actions in first person video. In *European Conference on Computer Vision (ECCV)*, pages 619–635, 2018.

[15] Yin Li, Zhefan Ye, and James M Rehg. Delving into egocentric actions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 287–295, 2015.

[16] Minlong Lu, Ze-Nian Li, Yueming Wang, and Gang Pan. Deep attention network for egocentric action recognition. *IEEE Transactions on Image Processing*, 28(8):3703–3713, 2019.

[17] Minghuang Ma, Haoqi Fan, and Kris M Kitani. Going deeper into first-person activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[18] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[19] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[20] Yuxin Peng, Yunzhen Zhao, and Junchao Zhang. Two-stream collaborative learning with spatial-temporal attention for video classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[21] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2847–2854, 2012.

[22] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[23] Suriya Singh, Chetan Arora, and CV Jawahar. First person action recognition using deep learned descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz. Lsta: Long short-term attention for egocentric action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9954–9963, 2019.

[25] Swathikiran Sudhakaran and Oswald Lanz. Attention is all we need: Nailing down object-centric attention for egocentric activity recognition. In *British Machine Vision Conference (BMVC)*, 2018.

[26] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[27] Min Tan, Baoyuan Wang, Zhaohui Wu, Jingdong Wang, and Gang Pan. Weakly supervised metric learning for traffic sign recognition in a lidar-equipped vehicle. *IEEE Transactions on Intelligent Transportation Systems*, 17(5):1415–1427, 2016.

[28] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497, 2015.

[29] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[30] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, 103(1):60–79, 2013.

[31] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision (ECCV)*, pages 20–36. Springer, 2016.

[32] Wenguan Wang, Jianbing Shen, Fang Guo, Ming-Ming Cheng, and Ali Borji. Revisiting video saliency: A large-scale benchmark and a new model. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[33] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[34] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *European Conference on Computer Vision (ECCV)*, pages 305–321, 2018.

[35] Jia Xu, Lopamudra Mukherjee, Yin Li, Jamieson Warner, James M Rehg, and Vikas Singh. Gaze-enabled egocentric video summarization via constrained submodular maximization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2235–2244, 2015.

[36] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 14, pages 77–81, 2015.

[37] Alfred L Yarbus. Eye movements during perception of complex objects. In *Eye Movements and Vision*, pages 171–211. 1967.

[38] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint pattern recognition symposium*, pages 214–223. Springer, 2007.

[39] Pengfei Zhang, Jianru Xue, Cuiling Lan, Wenjun Zeng, Zhanning Gao, and Nanning Zheng. Adding attentiveness to the neurons in recurrent neural networks. In *European Conference on Computer Vision (ECCV)*, pages 135–151, 2018.