

MODELING CROSS-VIEW INTERACTION CONSISTENCY FOR PAIRED EGOCENTRIC INTERACTION RECOGNITION

Zhongguo Li*, Fan Lyu*, Wei Feng*, Song Wang*[†]

*College of Intelligence and Computing, Tianjin University, China

[†]Department of Computer Science and Engineering, University of South Carolina, USA
{lizhongguo,fanlyu}@tju.edu.cn, wfeng@ieee.org, songwang@cec.sc.edu

ABSTRACT

With the development of Augmented Reality (AR), egocentric action recognition (EAR) plays an important role in accurately understanding demands from the user. However, EAR is designed to help recognize human-machine interaction in single egocentric view, thus difficult to capture interactions between two face-to-face AR users. Paired egocentric interaction recognition (PEIR) is the task to collaboratively recognize the interactions between two persons with the videos in their corresponding views. Unfortunately, existing PEIR methods always directly use linear decision function to fuse the features extracted from two corresponding egocentric videos, which ignore the consistency of interaction in paired egocentric videos. The consistency of interactions in paired videos, and features extracted from them, are correlated to each other. On top of that, we propose to derive the relevance between two views using bilinear pooling, which captures the consistency of two views in feature-level. Specifically, each neuron in the feature maps from one view connects to the neurons from the other view, which enforces the compact consistency between two views and then all possible paired neurons are used for PEIR. To be efficient, we use compact bilinear pooling with Count Sketch to avoid directly computing outer product. Experimental results on the PEV dataset shows the superiority of the proposed methods on the task PEIR.

Index Terms— Paired egocentric interaction recognition, bilinear pooling, action recognition

1. INTRODUCTION

Due to the advance of Augmented Reality (AR) techniques, wearable AR devices like Microsoft HoloLens allow users to interact with the real world via gestures or voice commands. Egocentric action recognition (EAR) [1, 2, 3, 4] is the task to recognize the action or gesture of users to achieve intelligent human-machine interaction. In this paper, we study the further problem of Paired Egocentric Interaction Recognition

This work was supported, in part, by the National Natural Science Foundation of China NSFC-U1803264, NSFC-61672376, and NSFC-61671325.

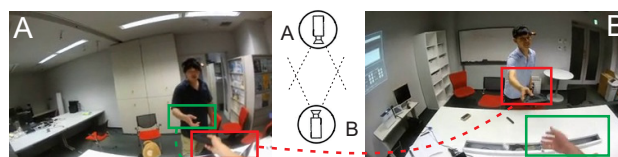


Fig. 1: An illustration of the consistency of interactions between a paired of egocentric views. Here the right hand of person A, labeled by red boxes, are recorded in both the views of A and B.

(PEIR) that recognizes the interactions between face-to-face AR users [5, 6, 7]. Different from EAR that only considers one egocentric video, PEIR needs to simultaneously consider the paired face-to-face egocentric videos. Utilizing paired egocentric videos recorded from face-to-face views can obtain more precise recognitions than egocentric videos from single views [8]. Enabling AR systems to understand the interactions between persons can provide more precise assistance in daily life. For examples, while one user point at one object and the other one shares his/her attention, AR system could read the interaction and response without explicit commands.

Previous works make many efforts on EAR [1, 2, 3, 4]. Ryoo et al. [4] tries to capture both entire scene dynamics and salient local motions observed in videos to predict interactions. Li et al. [1] tries to model the relationship of the camera wearer and the interactor. From EAR methods, a naive solution to PEIR is directly fusing the features from two views for the sake of one head output. For instance, one can concatenate the features extracted from paired egocentric videos and directly use the generated feature for linear classification. Till now, few works try to solve the PEIR problem. In [8], Yonetani et al. adopt a linear decision function to combine two kinds of handcrafted feature for PEIR.

In our view, the naive feature fusion from two views ignores the interaction consistency in two videos taken from two persons' views respectively. As shown in Fig. 1, we consider two persons of A and B and their paired views are shown as A and B respectively. On the one hand, the head's tilting of A could be recorded in the view of B, and also leads to the

shift of A’s viewpoints, because cameras are mounted over their heads. On the other hand, there may be interactions occurred in common areas recorded in both views of A and B. In the both cases, there are explicit or implicit information consistency of interactions between the views of A and B.

Due to the consistency of interactions in two views, neurons of feature maps corresponding to two views should describe the same information. We propose to classify interactions based on the consistent information represented by all possible paired neurons. Specifically, we use bilinear pooling, a second order polynomial kernel function [9], to capture the compact consistent information of interaction in all pairs. To avoid directly computing expensive outer product, we propose to use compact bilinear pooling to reduce the computation cost by using Count Sketch. We first extract features from paired egocentric videos, followed by obtaining sketch mentioned above and transforming them into Fourier domain. Finally we compute element-wise product of them and transform the result into real domain for linear classification. Experimental results on the PEV dataset shows that the proposed method outperforms other methods using naive fusions.

2. RELATED WORKS

Paired egocentric interaction recognition (PEIR), extended from egocentric action recognition (EAR), aims at recognizing the interaction between two face-to-face persons from both of their views. Along with the successes of deep learning in image-level tasks [10, 11], in recent years, EAR makes a great progress by using deep neural networks. For example Li et al. [1] models the relationship of the camera wearer and the interactor. However, EAR assumes there exists only one person with wearable camera. PEIR was first proposed by Yonetani et al. [8], where interactions like subtle motion of head or small hand actions people used in communications are recognized, by using their paired face-to-face views. The video from each view is collected by the camera over the head. To recognize interaction, Yonetani et al. [8] combines two kinds of hand-crafted features – PoTCD [12] for head and iDT [13] for body – for two views respectively with linear decision models. However, it ignores the consistency of interactions in two views and prefers to rely on only one view for prediction. Different from naive fusion like concatenation, we propose to leverage bilinear pooling to model the consistency between the paired egocentric videos.

Bilinear pooling models [14] was proposed for fine-grained image classification. Bilinear methods have been used to fuse two kind of features extracted by deep neural networks. Given two features $\mathbf{f}^1, \mathbf{f}^2 \in \mathbb{R}^C$, bilinear methods compute the outer product of them by

$$\mathbf{f}^1 \otimes \mathbf{f}^2 = [\mathbf{f}^1 \times (\mathbf{f}^2)^T] \in \mathbb{R}^{C^2}, \quad (1)$$

where \otimes means outer product of tensors. $[\cdot]$ operator vectorizes a matrix into a vector which means elements in matrix is

sorted by orders in new vector. Then the features generated by bilinear methods are directly used for classification. However when C is large, the generated features are of dimension C^2 . High-dimensional features make the direct computing of the outer product for linear classification very expensive. Gao et al. [9] reduces the dimension of generated feature from C^2 into D , which is far less than C^2 . Following this idea, Fukui et al. [15] fuse the visual features and the text features for visual question answering and visual grounding.

3. METHOD

3.1. Problem formulation

Denote the person who starts the interaction as A, the one who receives the interaction as B, as shown in Fig. 1 and their recorded videos (taken by the wearable cameras mounted over their heads) as v^A and v^B , respectively. PEIR tries to learn to recognize the interaction l^* from such paired egocentric videos based on the hypothesis $l^* = h(v^A, v^B)$.

The naive method to tackle PEIR is directly fusing the information from two views, such as [8], and this method can be formulated as

$$\mathbf{f}^A = \text{CNN}(v^A), \quad (2)$$

$$\mathbf{f}^B = \text{CNN}(v^B), \quad (3)$$

$$\mathbf{g} = \Phi(\mathbf{f}^A, \mathbf{f}^B), \quad (4)$$

$$p(l|v^A, v^B) = \sigma(\mathbf{W}^\top \mathbf{g} + \mathbf{b}), \quad (5)$$

where $\mathbf{f}^A, \mathbf{f}^B \in \mathbb{R}^C$ are the flattened feature maps extracted by Convolutional Neural Network (CNN) and $\mathbf{g} \in \mathbb{R}^D$ is the fused feature of \mathbf{f}^A and \mathbf{f}^B , σ is the softmax activation function. The key step is the design of the fusing function Φ to effectively combine the information from two views. Common choices of the fusing function Φ include

- 1) concatenation: $\Phi(\mathbf{x}, \mathbf{y}) = [\mathbf{x}, \mathbf{y}]$,
- 2) element-wise summation: $\Phi(\mathbf{x}, \mathbf{y}) = \mathbf{x} + \mathbf{y}$, and
- 3) element-wise product: $\Phi(\mathbf{x}, \mathbf{y}) = \mathbf{x} \odot \mathbf{y}$.

These three common fusing methods are easy to implement but do not consider the possible interaction consistency between two views. In this following, we propose a new method to address this problem.

3.2. Main Method

The consistency of interactions exist in paired egocentric videos as the common visual information of co-viewed objects, i.e., \mathbf{f}^A and \mathbf{f}^B are correlated. We first enumerate all pairs between \mathbf{f}^A and \mathbf{f}^B by

$$\mathbf{g}_k = \phi(\mathbf{f}_i^A, \mathbf{f}_j^B), \quad (6)$$

where $\mathbf{g} \in \mathbb{R}^{C^2}$ represents the consistent information in the form of pairs and $k = i \cdot C + j$ represents the k -th pair be-

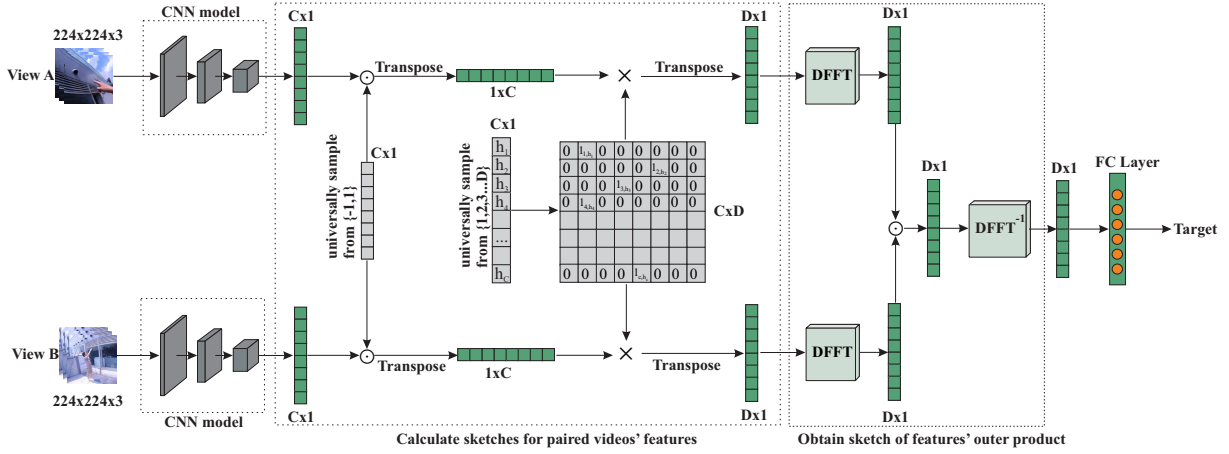


Fig. 2: Framework of the proposed method. Given paired egocentric videos from views A and B, we first extract features from both videos, then calculate the sketches of them, finally we obtain the sketch of outer product and transform it into Fourier domain for classification. DFFT is the discrete fast Fourier transform.

tween \mathbf{f}^A and \mathbf{f}^B . f_i is the i -th element in \mathbf{f} and ϕ is element-wise form of Φ . By Eq. (6), we construct element-wise information correlation between view A and view B, which yields fine-grained interaction consistency representation \mathbf{g} . In Eq. (6), if the product operator is adopted for ϕ , \mathbf{g} is actually the outer product of \mathbf{f}^A and \mathbf{f}^B , and Φ becomes the bilinear method as described in Eq. (1). The procedure of the compact bilinear method is shown in Alg. 1 in the appendix. To reduce the computation cost, we use the compact bilinear method [16, 9, 15] to shrink the dimension of \mathbf{g} from C^2 to D , where $D \ll C^2$, and the full framework is shown in Fig. 2. Gao et al. [9] proves that bilinear method is actually a second order polynomial kernel function and uses Count Sketch (called sketch in the remainder of this paper) [9, 17] to reduce the computation cost.

Detailedly, we first calculate the sketch of feature \mathbf{f}^A and \mathbf{f}^B . We define \mathbf{h}, \mathbf{s} as following: each element in $\mathbf{h} \in \mathbb{N}_+^C$ is universal randomly sampled from $\{1, 2, 3, \dots, D\}$ and every element in $\mathbf{s} \in \mathbb{Z}^C$ is universal randomly sampled from $\{-1, 1\}$. We denote \mathbf{H} as a $C \times D$ matrix generated according to \mathbf{h} : each element $\mathbf{H}_{i,j}$ is defined as

$$\mathbf{H}_{i,j} = \begin{cases} 1, & \text{if } j = \mathbf{h}_i, \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

For the feature \mathbf{f} , we first compute element-wise product of \mathbf{f} and \mathbf{s} , similar to traditional Random Maclaurin [18] for approximating the polynomial kernel. Then we compute matrix product of the above element-wise product and matrix \mathbf{H} to get sketch of feature \mathbf{f} , which projects \mathbf{f} from \mathbb{R}^C into \mathbb{R}^D to provide bounds on the variance of estimates to guarantee the reliability [16]:

$$\text{sketch}^A = (\mathbf{s}^A \odot \mathbf{f}^A) \mathbf{H}^A, \quad (8)$$

$$\text{sketch}^B = (\mathbf{s}^B \odot \mathbf{f}^B) \mathbf{H}^B. \quad (9)$$

After that, we transform sketches into Fourier domain using discrete fast Fourier transform (DFFT) and obtain element-wise product result of two sketches in Fourier domain. It has been proved [16] that the sketch of outer product $\mathbf{f}^A \otimes \mathbf{f}^B$ in Fourier domain is just the element-wise product of \mathbf{f}^A 's sketch sketch^A and \mathbf{f}^B 's sketch sketch^B both in Fourier domain, and the computation of outer product could be replaced by computing element-wise product in Fourier domain. Finally we obtain \mathbf{g} in the original domain by transforming the result in Fourier domain back using inverse DFFT (DFFT^{-1}), i.e.,

$$\mathbf{g} = \text{DFFT}^{-1}(\text{DFFT}(\text{sketch}^A) \odot \text{DFFT}(\text{sketch}^B)). \quad (10)$$

3.3. Discussion

Although the concatenation method and element-wise operations can also be written in the form of Eq. (2), they cannot describe the consistency of interaction in two views. The reasons are discussed as follows.

Concatenation. For concatenation methods, the fusing function Φ in Eq. (4) is to concatenate \mathbf{f}^A and \mathbf{f}^B , and the logits of the PEIR prediction can be written as

$$p(l|v^A, v^B) \propto \prod_{i=1}^{2C} \exp(\mathbf{W}_i^\top \mathbf{g}_i), \quad (11)$$

$$= \prod_{i=1}^C \exp(\mathbf{W}_i^\top \mathbf{f}_i^A) \prod_{i=C+1}^{2C} \exp(\mathbf{W}_i^\top \mathbf{f}_i^B). \quad (12)$$

Since the first C elements of $\mathbf{g} \in \mathbb{R}^{2C}$ generated by the concatenation method are all from feature \mathbf{f}^A and the rest are all from \mathbf{f}^B , we could rewrite the predicted probability of the model defined in Eq. (5) as Eq. (12). Obviously, in Eq. (12), the feature \mathbf{f}^A is independent of \mathbf{f}^B for final prediction. Thus concatenation method does not consider the correlation between \mathbf{f}^A and \mathbf{f}^B .

Element-wise product. Element-wise product is similar to Eq. (6), and the predicted probability could be written as

$$p(l|v^A, v^B) \propto \prod_{i=1}^C \exp(\mathbf{W}_i^\top \mathbf{f}_i^A \mathbf{f}_i^B), \quad (13)$$

where only feature elements with the same index is correlated.

Element-wise summation. If we adopt element-wise summation as ϕ , we could see

$$\begin{aligned} p(l|v^A, v^B) &\propto \prod_{i=1}^C \exp(\mathbf{W}_i^\top (\mathbf{f}_i^A + \mathbf{f}_i^B)) \\ &= \prod_{i=1}^C \exp(\mathbf{W}_i^\top \mathbf{f}_i^A) \prod_{i=1}^C \exp(\mathbf{W}_i^\top \mathbf{f}_i^B). \end{aligned} \quad (14)$$

Similar to the concatenation-based fusion, in this case features \mathbf{f}^A and \mathbf{f}^B are independent of each other for final prediction.

Ours. We adopt bilinear pooling operation as Φ , which means the product operator is selected as ϕ in Eq. ((6)), i.e.,

$$p(l|v^A, v^B) \propto \prod_{i=1, j=1}^C \exp(\mathbf{W}_i^\top \mathbf{f}_i^A \mathbf{f}_j^B), \quad (15)$$

where the logits of PEIR are predicted based on the paired elements between \mathbf{f}^A and \mathbf{f}^B . Each element in \mathbf{f}^A is correlated with all the elements in \mathbf{f}^B for final prediction.

Table 1: Accuracy in validation set with RGB or OpticalFlow.

Modality	View	Method	D	Split1	Split2	Split3	Avg Acc
RGB	A	-	-	78.50	77.94	76.19	77.54
RGB	B	-	-	73.52	68.73	71.75	71.33
RGB	A+B	Avg	-	80.50	80.55	76.80	79.28
RGB	A+B	Svm	-	84.96	78.70	80.40	81.35
RGB	A+B	Concat	2048	82.25	80.89	79.30	80.81
RGB	A+B	Ours	2048	84.35	83.09	83.47	83.63
OF	A	-	-	84.96	81.35	84.24	83.51
OF	B	-	-	79.74	78.00	78.46	78.73
OF	A+B	Avg	-	86.01	82.62	85.86	84.83
OF	A+B	Svm	-	89.40	83.42	87.97	86.93
OF	A+B	Concat	2048	87.37	87.56	88.21	87.71
OF	A+B	Ours	2048	88.10	89.06	89.12	88.76
RGB	A+B	Sum	1024	82.99	79.14	79.09	80.41
RGB	A+B	Product	1024	84.35	79.10	72.81	78.75
RGB	A+B	Ours	1024	86.08	81.34	83.78	83.73
OF	A+B	Sum	1024	87.24	87.26	87.64	87.78
OF	A+B	Product	1024	87.30	88.44	88.34	88.03
OF	A+B	Ours	1024	89.33	88.81	88.37	88.84

4. EXPERIMENTS

4.1. Experiments details

Dataset and evaluation metric. PEV dataset is proposed in [8], which contains 1,226 pairs of videos in 7 categories. Each paired video is collected by the cameras

Table 2: Mean accuracy of TwoStream nets.

Modality	View	Method	D	Avg Acc
PoTCD+IDT	A+B	Yonetani et al.[8]	-	69.2
TwoStream	A+B	Svm	-	87.27
TwoStream	A+B	Concat	2048	87.72
TwoStream	A+B	Ours	2048	88.91
TwoStream	A+B	Product	1024	85.17
TwoStream	A+B	Sum	1024	84.41
TwoStream	A+B	Ours	1024	88.45



Fig. 3: Confusion matrix of our method, averaged over the three validation sets.

mounted to the heads of two persons standing face to face. The 7 categories of interactions present in PEV dataset[8] are: *Pointing, Attention, Positive, Negative, Passing, Receiving, Gesture*. We adopt the three-fold cross validation for evaluation. Results in all three validation splits are reported in Table 1.

Model settings. We adopt I3D [19] as the backbone, which is pretrained on Kinetic-400. C , the dimension of \mathbf{f}^A and \mathbf{f}^B , is set to 1,024 as used in backbone. For fair comparison, D , the dimension of \mathbf{g} , is set to 2,048 as in the concatenation method, or 1,024 as in the element-wise product or summation methods. Cross entropy loss is chosen as the loss function. Standard SGD is used to train neural networks in this paper, and the learning rate is set to 0.1.

Normalization. \mathbf{f}^A and \mathbf{f}^B shall be normalized to be in the form of one-hot encoding, since they are batch normalized [20] in the backbone and fed into ReLU [21] layer. We only scale them by a constant factor.

Data Processing. We sample one frame for every three frames. Sampled frames are randomly cropped and scaled to 224×224 . Horizontal flipping is then used for data augmentation. The Length of clip is set to 32, and the last frame will be repeated if clip's length is less than 32.

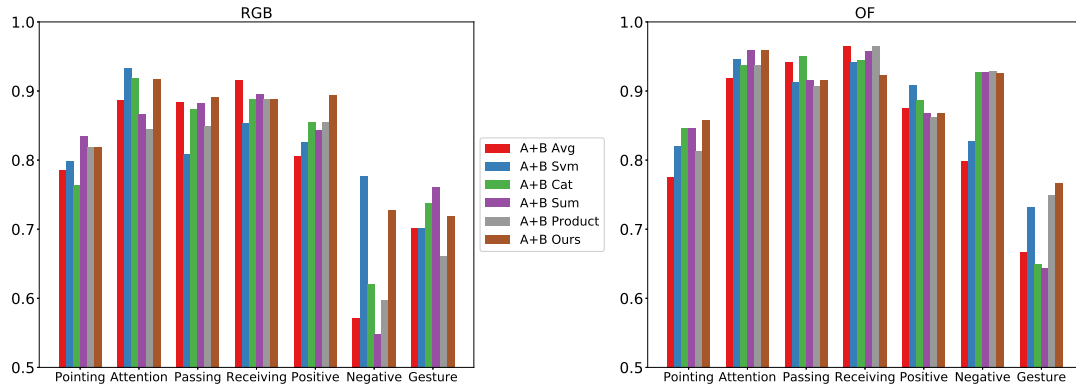


Fig. 4: Accuracy for each category while input is RGB (left) or OpticalFlow (right). ‘A+B’ means both views are used.

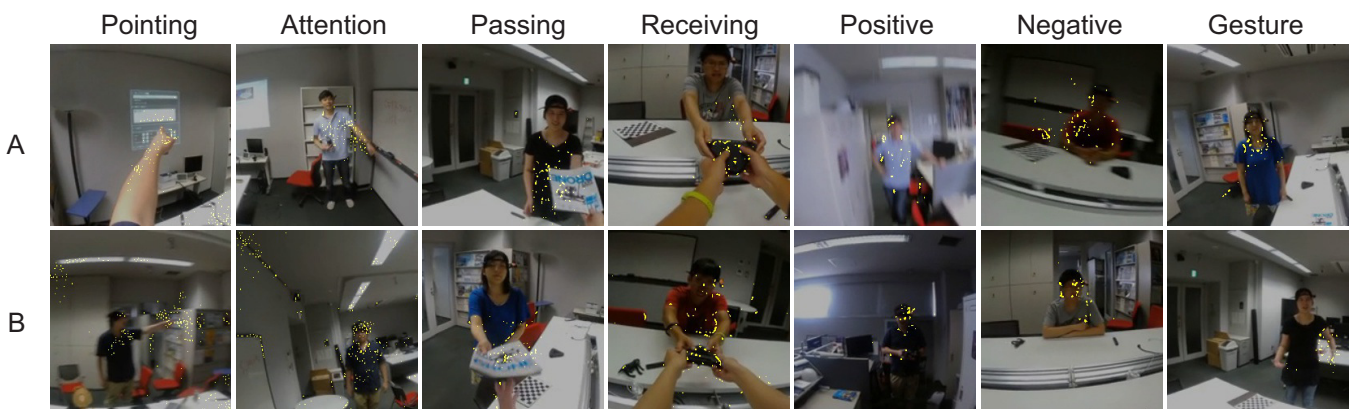


Fig. 5: Top 10 percent integrated gradients of our model in validation sets. The yellow dots show the model’s focuses.

4.2. Results

We explore different fusion methods under RGB or OpticalFlow modality. We choose feature-fusion methods including concatenation (Concat), element-wise product (Product) and element summation (Sum) as baselines. We also report results of the methods using average (Avg) or SVM for late-fusion, i.e., training classifier for each view first and then fusing scores as the final result. Results are shown in Table 1.

Comparisons. All the methods using two views together outperform the models that only use one view. We can see that, SVM is always a better choice for fusing decision scores. Element-wise product achieves a score higher than element-summation in OpticalFlow modality, but lower in RGB modality. While under RGB modality, the end-to-end concatenation method does not outperform the score fusion method, e.g., SVM, the result is opposite when the input becomes OpticalFlow. Under OpticalFlow modality, element-wise product is surprisingly better than concatenation or element-wise summation, with half parameters in the last fully connected layer compared with the concatenation-

based method. Our method performs best in both RGB and OpticalFlow modalities – it outperforms the second best method by 2.28% in RGB modality, and by 1.05% in OpticalFlow modality, with D set as 2,048. When D is set to 1,024, our proposed method outperforms the second best by 3.32% in RGB modality, and by only 0.81% (over element-wise product) in OpticalFlow modality. A possible explanation is that only motion information is recorded in OpticalFlow modality and the consistency involving appearance cannot be built in OpticalFlow modality. When following the two-stream net [22] of using both RGB and OpticalFlow modalities as input, our method still achieves highest score than other fusing methods, as shown in Table. 2.

Confusion matrix and category-grained accuracy. Confusion matrix of our method is shown in Fig. 3 and Fig. 4. The proposed method performs best or pretty close to the best. Other methods is unstable compared with ours. We believe this is because our proposed method makes final prediction only if evidence is found in both paired egocentric videos. For RGB modality, *Negative* action is the most difficult to be recognized, where the score-fusion SVM method performs

the best, followed by our method as the second. For OpticalFlow modality, the most difficult action is *Gesture* and the proposed method achieves the highest score.

4.3. Visualization

We use Integrated Gradients [23] to visualize the model, as shown in Fig. 5. The yellow dots in frames represent the focus of the model. In *Pointing* category, person A's hand is focused, and other dots in the edges of frame represent the model focus on these pixels to catch shift of attentions, and similar results can be seen in *Attention* category. The visualization for *Passing* and *Receiving* actions shows that the model successfully focuses on the hand and object. For *Positive* and *Negative* actions, the focus is changed into A and B's upper body. The interesting result occurs for *Gesture* action, the model focuses on B's head, possibly because B responds to A with a subtle shaking of head.

5. CONCLUSION

In this paper, we proposed to build the relevance between paired egocentric videos for interaction recognition. We observed that the consistency of interactions exists in paired videos and features extracted from them are correlated to each other. We proposed to use bilinear pooling to capture all possible consistent information represented in pairs of elements between the features from two views. Moreover we used the compact bilinear pooling with Count Sketch to reduce the algorithm computation complexity. Experiments showed that our method achieves the state-of-the-art performance on the PEV dataset.

6. REFERENCES

- [1] Haoxin Li, Yijun Cai, and Wei-Shi Zheng, "Deep dual relation modeling for egocentric interaction recognition," in *CVPR*, 2019.
- [2] Vasileios Choutas, Philippe Weinzaepfel, Jerome Revaud, and Cordelia Schmid, "Potion: Pose motion representation for action recognition," in *CVPR*, 2018.
- [3] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz, "LSTA: Long short-term attention for egocentric action recognition," in *CVPR*, 2019.
- [4] Michael Ryoo, Brandon Rothrock, and Larry Matthies, "Pooled motion features for first-person videos," in *CVPR*, 2015.
- [5] Mohammad Syahputra, Siti Fatimah, and Romi Rahmat, "Interaction on Augmented Reality with Finger Detection and Hand Movement Recognition," in *AVR*, 2018.
- [6] Ahmad Karambakhsh, Aouaidjia Kamel, Bin Sheng, Ping Li, Po Yang, and David Feng, "Deep gesture interaction for augmented anatomy learning," *International Journal of Information Management*, vol. 45, pp. 328–336, APR 2019.
- [7] Xueni Pan and Antonia Hamilton, "Why and how to use virtual reality to study human social interaction: The challenges of exploring a new research landscape," *British Journal of Psychology*, vol. 109, no. 3, pp. 395–417, AUG 2018.
- [8] Ryo Yonetani, Kris M Kitani, and Yoichi Sato, "Recognizing micro-actions and reactions from paired egocentric videos," in *CVPR*, 2016.
- [9] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell, "Compact bilinear pooling," in *CVPR*, 2016.
- [10] Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan, "Visual Grounding via Accumulated Attention," in *CVPR*, 2018.
- [11] Fan Lyu, Qi Wu, Fuyuan Hu, Qingyao Wu, and Mingkui Tan, "Attend and Imagine: Multi-Label Image Classification With Visual Attention and Recurrent Neural Networks," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 1971–1981, AUG 2019.
- [12] Yair Poleg, Chetan Arora, and Shmuel Peleg, "Temporal segmentation of egocentric videos," in *CVPR*, 2014.
- [13] Heng Wang and Cordelia Schmid, "Action recognition with improved trajectories," in *ICCV*, 2013.
- [14] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji, "Bilinear cnn models for fine-grained visual recognition," in *CVPR*, 2015.
- [15] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach, "Multimodal compact bilinear pooling for visual question answering and visual grounding," in *EMNLP*, 2016.
- [16] Ninh Pham and Rasmus Pagh, "Fast and scalable polynomial kernels via explicit feature maps," in *SIGKDD*, 2013.
- [17] Rasmus Pagh, "Compressed matrix multiplication," in *ITCS*, 2012.
- [18] Purushottam Kar and Harish Karnick, "Random feature maps for dot product kernels," in *AISTATS*, 2012.
- [19] Joao Carreira and Andrew Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *CVPR*, 2017.
- [20] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.
- [21] Xavier Glorot, Antoine Bordes, and Yoshua Bengio, "Deep sparse rectifier neural networks," in *AIS*, 2011.
- [22] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*, 2014.
- [23] Mukund Sundararajan, Ankur Taly, and Qiqi Yan, "Ax- iomatic attribution for deep networks," in *ICML*, 2017.