Deep Dual Relation Modeling for Egocentric Interaction Recognition

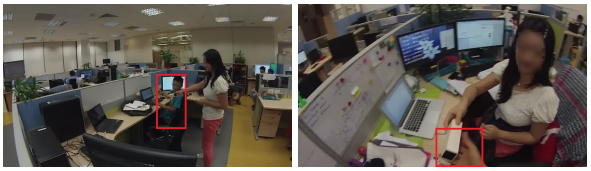
Haoxin Li[1,3,4], Yijun Cai[1,4], Wei-Shi Zheng[2,3,4,*]
[1]School of Electronics and Information Technology, Sun Yat-sen University, China
[2]School of Data and Computer Science, Sun Yat-sen University, China
[3]Peng Cheng Laboratory, Shenzhen 518005, China
[4]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
lihaoxin05@gmail.com, caiyj6@mail2.sysu.edu.cn, wszheng@ieee.org

Abstract

Egocentric interaction recognition aims to recognize the camera wearer's interactions with the interactor who faces the camera wearer in egocentric videos. In such a human-human interaction analysis problem, it is crucial to explore the relations between the camera wearer and the interactor. However, most existing works directly model the interactions as a whole and lack modeling the relations between the two interacting persons. To exploit the strong relations for egocentric interaction recognition, we introduce a dual relation modeling framework which learns to model the relations between the camera wearer and the interactor based on the individual action representations of the two persons. Specifically, we develop a novel interactive LSTM module, the key component of our framework, to explicitly model the relations between the two interacting persons based on their individual action representations, which are collaboratively learned with an interactor attention module and a global-local motion module. Experimental results on three egocentric interaction datasets show the effectiveness of our method and advantage over state-of-the-arts.

1. Introduction

Egocentric interaction recognition [11, 25, 31, 35, 39] attracts increasing attention with the popularity of wearable cameras and broad applications including human machine interaction [2, 18] and group events retrieval [3, 4]. Different from exocentric (third-person) videos, in egocentric videos, the camera wearers are commonly invisible and the videos are usually recorded with dynamic ego-motion (see Figure 1). The invisibility of the camera wearer hampers action recognition learning of the camera wearer, and the ego-motion hinders direct motion description of the interactor, which make egocentric interaction recognition challenging.



(a) Invisibility of the camera wearer

(b) Ego-motion of the camera wearer

Figure 1. Illustration of camera-wearer's invisibility and ego-motion. (a) compares the person (in red boxes) receiving something in exocentric (left) and egocentric (right) videos from NUSF-PID dataset [25]. (b) shows adjacent frames with obvious ego-motion in an egocentric video from UTokyo PEV dataset [39].

An egocentric interaction comprises the actions of the camera wearer and the interactor that influence each other with relations. So modeling the relations between the two interacting persons is important for interaction analysis. To model the relations between the the two interacting persons explicitly, we need to obtain individual action representations of the two persons primarily. Therefore, we formulate the egocentric interaction recognition problem into two interconnected subtasks, individual action representation learning and dual relation modeling.

In recent years, various works attempt to recognize interactions from egocentric videos. Existing methods integrated motion information using statistical properties of trajectories and optical flows [25, 31, 38] or utilized face orientations descriptors [11] with SVM classifiers for recognition. Deep neural network was also adopted to aggregate short-term and long-term information for classification [35]. However, some of them [31] aimed to recognize in-

*Corresponding author

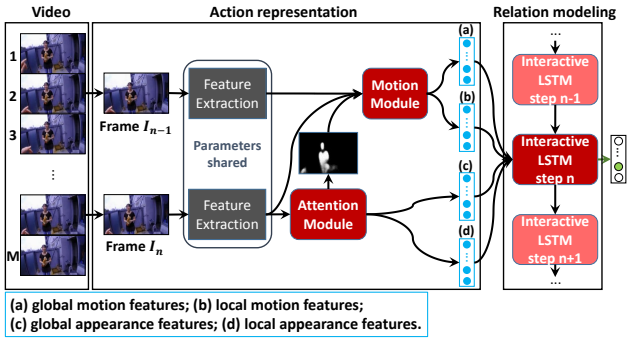


Figure 2. Proposed framework. Frames $\boldsymbol{I}_i (i = 1, ..., N)$ are sampled from the video as input. The Feature Extraction Module extracts basic visual features of sampled frames. The Attention Module localizes the interactor and learns appearance features. The Motion Module estimates global and local motions for motion features learning. The Interaction Module models the relations for better interaction recognition based on the learned individual features (a), (b), (c) and (d) explained in the blue box.

teractions from a static observer's view, which is impractical for most applications. Others [11, 25, 35] directly learned interaction as a whole through appearance and motion learning as done in common individual action analysis. They didn't learn individual action representations of the interacting persons, and thus failed to model the relations explicitly. The first-person and second-person features were introduced to represent the actions of the camera wearer and the interactor in [39]. But they learned the individual action representations from multiple POV (point-of-view) videos and still lacked explicit relation modeling.

Overview of the framework. In this paper, we focus on the problem of recognizing human-human interactions from single POV egocentric videos. Considering the relations in egocentric interactions, we develop a dual relation modeling framework, which integrates the two interconnected subtasks, namely individual action representation learning and dual relation modeling, for recognition as shown in Figure 2. Specifically, for dual relation modeling, we develop an interaction module, termed interactive LSTM, to model the relations between the camera wearer and the interactor explicitly based on the learned individual action representations. For individual action representations learning, we introduce an attention module and a motion module to jointly learn action features of the two interacting persons. We finally combine these modules into an end-to-end framework and train them with human segmentation loss, frame reconstruction loss and classification loss as supervision. Experimental results indicate the effectiveness of our method.

Our contributions. In summary, the main contribution of this paper is three-fold. (1) An interactive LSTM module is developed to model the relations between the camera wearer and the interactor from single POV egocentric videos. (2) An interactor attention module and a global-local motion module are designed to jointly learn individual action representations of the camera wearer and the interactor from single POV egocentric video. (3) By integrating individual action representations learning and dual relation modeling into an end-to-end framework, our method shows its effectiveness and outperforms existing state-of-the-arts on three egocentric interaction datasets.

2. Related Work

Egocentric action recognition aims to recognize camera wearer's actions from first-person videos. Since the egomotion is a dominant characteristic of egocentric videos, most methods used dense flow or trajectory based statistical features [1, 13, 17, 23, 24] to recognize the actions of the camera wearer. In some object-manipulated actions, some works extracted hands and objects descriptors for recognition [10, 20, 28, 43], and others further explored gaze information according to hand positions and motions [12, 19]. Recently, deep neural networks have also been applied to egocentric action recognition. Frame-based feature series analysis showed their promising results [16, 32, 40]. CNN networks with multiple information streams were also trained on recognition task [22, 34]. However, these methods target on individual actions which are a bit different from human-human interactions.

Egocentric interaction recognition specifically focuses on first-person human-human interactions. Ryoo *et al*. recognized what the persons in the videos are doing to the static observer [30, 31], but it is unrealistic in most daily life scenarios. Some works used face orientations, individual locations descriptors and hand features to recognize interactions [5, 11]. Others used motion information based on the magnitudes or clusters of trajectories and optical flows [25, 38]. A convLSTM was utilized to aggregate features of successive frames for recognition [35]. These methods commonly learned interaction descriptors by direct appearance or motion learning, but didn't considered explicit relation modeling with individual action representations of the camera wearer and the interactor. Yonetani *et al*. learned individual action features of the two persons but also lacked explicit relations modeling [39].

Different from the existing methods above, our framework jointly learns individual actions of the camera wearer and the interactor from single POV egocentric videos and further explicitly models the relations between them by an interactive LSTM.

3. Individual Action Representation Learning

To model the relations, we first need individual action representations of the camera wearer and the interactor. Here, we learn interactor masks to separate the interactor from background and learn appearance features with an at-

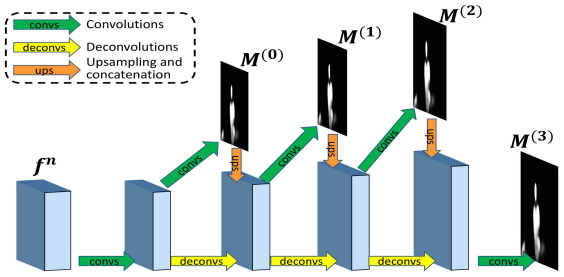


Figure 3. Structure of attention module. The module takes feature $\boldsymbol{f}^n$ as input and generates attention weighted features and multi-scale masks.

tention module. In the meanwhile, a motion module is integrated to learn motion cues, so that we can jointly learn the individual appearance and motion features of the two persons, which are the basis to model relations in Section 4.

For two consecutive sampled frames $\boldsymbol{I}_{n-1}, \boldsymbol{I}_n \in \mathbb{R}^{H\times W\times 3}$, we use a feature extraction module composed of ResNet-50 [14] to extract basic features $\boldsymbol{f}(\boldsymbol{I}_{n-1}), \boldsymbol{f}(\boldsymbol{I}_n) \in \mathbb{R}^{H_0\times W_0\times C}$, which encode the scene or human information with multidimensional representations for further modeling on top of them. In the following, we denote $\boldsymbol{f}(\boldsymbol{I}_{n-1})$ and $\boldsymbol{f}(\boldsymbol{I}_n)$ as $\boldsymbol{f}^{n-1}$ and $\boldsymbol{f}^n$ respectively for convenience.

3.1. Attention Appearance Features Learning

Egocentric videos record the actions of the camera wearer and the interactor simultaneously. To learn individual action features of the two persons, we wish to separate the interactor from background based on the feature $\boldsymbol{f}^n$.

Pose-guided or CAM-guided strategy [8, 9, 27] is used for person attention learning. Similarly, we introduce an attention module to localize the interactor with human segmentation guidance. We employ a deconvolution structure [26] on top of the basic feature $\boldsymbol{f}^n$ to generate the masks of the interactor as shown in Figure 3. Mask $\boldsymbol{M}^{(0)} \in \mathbb{R}^{H_0\times W_0}$ serves to weight the corresponding feature maps for attention features learning. Multi-scale masks $\boldsymbol{M}^{(k)} \in \mathbb{R}^{H_k\times W_k}(k=1,2,3)$ are applied to localize the interactor at different scales for finer masks generation and explicit motion estimation later in Subsection 3.2.

Mask of the Interactor. To localize the interactor, we introduce a human segmentation loss to guide the learning of our attention module. Given a reference mask $\boldsymbol{M}^{RF}$, the human segmentation loss is a pixel-wise cross entropy loss:

$$L_{seg} = -\sum_{k=1}^{3}\sum_{i=1}^{H_k}\sum_{j=1}^{W_k}\frac{1}{H_k\times W_k}[\boldsymbol{M}^{RF}_{i,j} log \boldsymbol{M}^{(k)}_{i,j} + (1-\boldsymbol{M}^{RF}_{i,j}) log(1-\boldsymbol{M}^{(k)}_{i,j})], \quad (1)$$

where k indexes the mask scales and the reference mask is resized to the corresponding shape for calculation. Here, the reference masks are obtained using JPPNet[21].

Attention Features. An optimized attention module could localize the interactor, so the mask $\boldsymbol{M}^{(0)}$ has higher values at the positions corresponding to the interactor, which indicates concrete appearance information of the interactor. Then the **l**ocal **a**ppearance feature describing the action of the interactor from its appearance can be calculated with weighted pooling as follows:

$$\boldsymbol{f}^n_{l,a} = \frac{1}{|\boldsymbol{M}^{(0)}|}\sum_{i=1}^{H_0}\sum_{j=1}^{W_0}\boldsymbol{M}^{(0)}_{i,j}\cdot \boldsymbol{f}^n_{i,j,1:C}, \quad (2)$$

where $|\boldsymbol{M}^{(0)}| = \sum_{i=1}^{H_0}\sum_{j=1}^{W_0}\boldsymbol{M}^{(0)}_{i,j}$. Accordingly, the **g**lobal **a**ppearance feature, which describes the action of the camera wearer from what is observed, is calculated using global average pooling:

$$\boldsymbol{f}^n_{g,a} = \frac{1}{H_0\times W_0}\sum_{i=1}^{H_0}\sum_{j=1}^{W_0}\boldsymbol{f}^n_{i,j,1:C}. \quad (3)$$

The attention module learns local appearance features to provide a concrete description instead of a global description of the interactor, and thus assists the relation modeling later. Meanwhile, the interactor masks localizing the interactor plays an important role in separating global and local motion, which is employed in Subsection 3.2.

3.2. Global-local Motion Features Learning

Motion features are vital for action analysis. To learn individual action representations of the two interacting persons, we wish to describe the ego-motion (global motion) of the camera wearer and the local motion of the interactor explicitly based on the basic features $\boldsymbol{f}^n$, $\boldsymbol{f}^{n-1}$ and the interactor masks $\boldsymbol{M}^{(k)}(k=0,1,2,3)$.

Differentiable warping scheme [15] is used for ego-motion estimation with a frame reconstruction loss [36, 42]. Inspired by them, we design a self-supervised motion module with the differentiable warping mechanism to jointly estimate the two types of motion from egocentric videos.

Global-local Motion Formulation by Reconstruction. To separate the global and local motions in egocentric videos, we reuse the interactor mask $\boldsymbol{M}^{(3)}$ generated in Subsection 3.1 with the same scale as the input frames to formulate the transformation between two adjacent frames. With the learnable parameters $\boldsymbol{T}$ and $\boldsymbol{D}$ denoting transformation matrix and dense motion field, we can formulate the transformation from homogeneous coordinates $\boldsymbol{X}_n$ to $\boldsymbol{X}_{n-1}$ concisely as:

$$\hat{\boldsymbol{X}}_{n-1} = \boldsymbol{T}(\boldsymbol{X}_n + \boldsymbol{M}^{(3)}\odot \boldsymbol{D}), \quad (4)$$

where $\odot$ is element-wise multiplication, $\boldsymbol{X}_n$ and $\boldsymbol{X}_{n-1}$ are homogeneous coordinates of frame $\boldsymbol{I}_n$ and $\boldsymbol{I}_{n-1}$.

In Equation (4), $\boldsymbol{M}^{(3)}\odot \boldsymbol{D}$ is the local dense motion field of the interactor, and $\boldsymbol{T}$ describes the ego-motion of

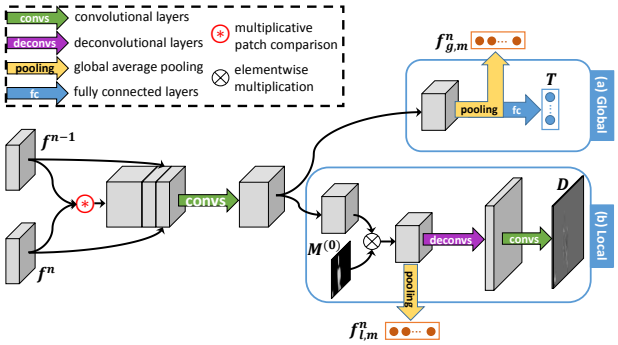


Figure 4. Structure of motion module. The module takes basic features $\boldsymbol{f}^n$, $\boldsymbol{f}^{n-1}$ and mask $\boldsymbol{M}^{(0)}$ as inputs and estimates global and local motion parameters in two branches in which global motion feature $\boldsymbol{f}^n_{g,m}$ and local motion feature $\boldsymbol{f}^n_{l,m}$ are extracted. $*$ in red circle is a multiplicative patch comparison [7] to calculate the correlations between two feature maps, which captures the relative motions between them for dense flow estimation.

the camera wearer, so Equation (4) jointly formulates global and local motion explicitly by point set reconstruction.

Self-supervision. To learn the parameters in Equation (4), we use view synthesis objective [42] as supervision:

$$L_{rec} = \sum_x |\boldsymbol{I}_n(x) - \hat{\boldsymbol{I}}_n(x)|, \tag{5}$$

where x indexes over pixel coordinates $\boldsymbol{X}_n$. And $\hat{\boldsymbol{I}}_n$ is the reconstructed frame warped from frame $\boldsymbol{I}_{n-1}$ according to the transformed point set $\hat{\boldsymbol{X}}_{n-1}$, which is employed with the bilinear sampling mechanism [15] as

$$\hat{\boldsymbol{I}}_n(x) = \sum_{i\in\{t,b\},j\in\{l,r\}} w^{ij}\boldsymbol{I}_{n-1}(\hat{x}^{ij}), \tag{6}$$

where $\hat{x}$ indexes over projected coordinates $\hat{\boldsymbol{X}}_{n-1}$, $\hat{x}^{ij}$ is the neighboring coordinate of $\hat{x}$, w^{ij} is proportional to the spatial proximity between $\hat{x}^{ij}$ and $\hat{x}$, and subject to $\sum_{i,j} w^{ij} = 1$. In addition, we regularize the local dense motions with a smoothness loss for robust learning [36].

With the reconstruction loss in Equation (5), we design a motion module illustrated in Figure 4 with two branches learning the parameters of global ego-motion and local motion in Equation (4), from which **g**lobal **m**otion feature $\boldsymbol{f}^n_{g,m}$ and **l**ocal **m**otion feature $\boldsymbol{f}^n_{l,m}$ are extracted from the embedding layers.

The motion module jointly estimates explicit motions of the camera wearer and the interactor by reusing the interactor masks, from which we learn concrete individual motion features of the two interacting persons hence aid relation modeling in Section 4.

3.3. Ego-feature and Exo-feature.

For each frame pair $\{\boldsymbol{I}_{n-1}, \boldsymbol{I}_n\}$, we obtain global appearance feature $\boldsymbol{f}^n_{g,a}$ and local appearance feature $\boldsymbol{f}^n_{l,a}$ from the attention module, and also global motion feature $\boldsymbol{f}^n_{g,m}$ and local motion feature $\boldsymbol{f}^n_{l,m}$ from the motion module. The global features describe overall scene context and ego-motion of the camera wearer, which could represent the action of the camera wearer. While the local features, obtained with the interactor masks, describe the concrete appearance and motion of the interactor, which could represent the action of the interactor. Thus we obtain the individual action representations of the two interacting persons.

For further exploring the relations between the two persons, we define the ***ego-feature*** $\boldsymbol{f}^n_{ego} = [\boldsymbol{f}^n_{g,a}, \boldsymbol{f}^n_{g,m}]$ describing the camera wearer, and ***exo-feature*** $\boldsymbol{f}^n_{exo} = [\boldsymbol{f}^n_{l,a}, \boldsymbol{f}^n_{l,m}]$ describing the interactor. With them we could model the relations in Section 4.

4. Dual Relation Modeling by Interactive LSTM

Given the action representations, a classifier may be trained for recognition as done in most previous works. However, as discussed before, a distinguishing property of egocentric human-human interactions is the relations between the camera wearer and the interactor, which deserves further exploration for better interaction representations.

We notice that only the ego-feature or exo-feature may not exactly represent an interaction. For the example shown in Figure 6, two interactions consist of similar individual actions: the camera wearer turning his head and the interactor pointing somewhere. In this case, neither the features of any action can identify an interaction sufficiently. However, some relations would clearly tell the differences of the two interactions, such as the sequential orders and the motion directions of the individual actions. To utilize the relations for recognition, we develop an interaction module to model the relations between the two persons based on the ego-feature and exo-feature defined in Subsection 3.3.

4.1. Symmetrical Gating and Updating

To model the relations such as the synchronism or complementarity between the two interacting persons, we integrate their action features using LSTM structure.

We define ***ego-state*** $\boldsymbol{F}^n_{ego}$ and ***exo-state*** $\boldsymbol{F}^n_{exo}$ to denote the latent states till the n-*th* step to encode the evolution of the two actions, which correspond to ego-feature and exo-feature introduced in Subsection 3.3, respectively. We wish to mutually incorporate the action context of each interacting person at each time step to explore the relations such as the synchronism and complementarity. Thus, we utilize exo-state to filter out the irrelevant parts, enhance the relevant parts and complement the absent parts of the ego-state. Meanwhile, the exo-state is also filtered, enhanced and complemented by the ego-state. This symmetrical gating and updating mechanism is realized with two symmet-

rical LSTM blocks where each block works as follows:

$$[\boldsymbol{i}^n;\boldsymbol{o}^n;\boldsymbol{g}^n;\boldsymbol{a}^n] = \sigma(\boldsymbol{W}\boldsymbol{f}^n + \boldsymbol{U}\boldsymbol{F}^{n-1} + \boldsymbol{J}^{n-1} + \boldsymbol{b}), \tag{7}$$

$$\boldsymbol{J}^n = \phi(\boldsymbol{V}\boldsymbol{F}_*^n + \boldsymbol{v}), \tag{8}$$

$$\boldsymbol{c}^n = \boldsymbol{i}^n\boldsymbol{a}^n + \boldsymbol{g}^n\boldsymbol{c}^{n-1}, \tag{9}$$

$$\boldsymbol{F}^n = \boldsymbol{o}^n tanh(\boldsymbol{c}^n). \tag{10}$$

Here, the input gate, output gate, forget gate and update candidate are denoted as $\boldsymbol{i}^n$, $\boldsymbol{o}^n$, $\boldsymbol{g}^n$ and $\boldsymbol{a}^n$ respectively. σ is tanh activation function for update candidate and sigmoid activation function for other gates. $\boldsymbol{F}_*$ is the latent state from the dual block, ϕ is ReLU activation function, and $\boldsymbol{J}^n$ is the modulated dual state. $\{\boldsymbol{W}, \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{b}, \boldsymbol{v}\}$ are parameters of each LSTM block.

It is noted that the current ego-state integrates the historical information of the ego-actions and also the exo-actions into itself, and vice versa for exo-state. The ego-state and exo-state describe the interaction from the view of the camera weearer and the interactor respectively. In this symmetrical gating and updating manner, the symmetrical LSTM blocks model the interactive relations instead of a raw combination of two actions.

4.2. Explicit Relation Modelling

Besides the symmetrical LSTM blocks introduced above for implicitly encoding the dual relations into the ego-state and exo-state, we further explicitly model the dual relation. To this end, we introduce relation-feature $\boldsymbol{r}^n$ to explicitly calculate the relations with a nonlinear additive operation on the ego-state and exo-state:

$$\boldsymbol{r}^n = tanh(\boldsymbol{F}_{ego}^n + \boldsymbol{F}_{exo}^n). \tag{11}$$

With the relation-feature $\boldsymbol{r}^n$ at each time step, we further model the time variant relations with another LSTM branch to integrate the historical relations into the relation-states $\boldsymbol{R}^n$, which can be formulated as follows:

$$[\boldsymbol{i}^n;\boldsymbol{o}^n;\boldsymbol{g}^n;\boldsymbol{a}^n] = \sigma(\boldsymbol{W}\boldsymbol{r}^n + \boldsymbol{U}\boldsymbol{R}^{n-1} + \boldsymbol{b}), \tag{12}$$

$$\boldsymbol{c}^n = \boldsymbol{i}^n\boldsymbol{a}^n + \boldsymbol{g}^n\boldsymbol{c}^{n-1}, \tag{13}$$

$$\boldsymbol{R}^n = \boldsymbol{o}^n tanh(\boldsymbol{c}^n). \tag{14}$$

In the equations above, the gates and parameters are similarly denoted as those in the symmetrical LSTM blocks. In Equation (14), $\boldsymbol{R}^n$ integrates historical and current relations information to explicitly represent the relations of the two actions at n-th time step during the interaction.

Combining the two components above, *i.e.* the symmetrical LSTM blocks and the relation LSTM branch, our interaction module is illustrated in Figure 5, which we term interactive LSTM. It captures the evolution or synchronism of the two actions and further explicitly models the relations between the two actions, which provides a better representation of the interaction.

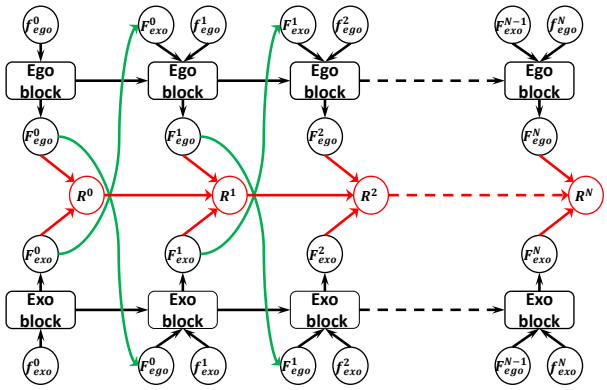


Figure 5. Diagram of Interactive LSTM. The unrolled symmetrical LSTM blocks mutually gate and update each other as the green arrows depict. The unrolled relation LSTM branch is highlighted in red. All LSTM blocks contains N time steps.

The posterior probability of an interaction category given the final relation-state $\boldsymbol{R}^N$ can be defined as

$$p(\boldsymbol{y}|\boldsymbol{R}^N) = \delta(\boldsymbol{W}\boldsymbol{R}^N + \boldsymbol{b}), \tag{15}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are parameters of classifier and δ is softmax function. Then a cross entropy loss function is employed to supervise parameters optimization as follow:

$$L_{cls} = -\sum_{k=1}^{K} y_k log[p(y_k|\boldsymbol{R}^N)], \tag{16}$$

where K is the number of class.

Combining the loss functions of each module above, we train our model end-to-end with the final objective:

$$L_{final} = L_{cls} + \alpha L_{seg} + \beta L_{rec} + \gamma L_{smooth}, \tag{17}$$

where α, β, γ are weights of segmentation loss, frame reconstruction loss and smooth regularization, respectively.

5. Experiments

5.1. Datasets

We evaluate our method on three egocentric human-human interaction datasets.

UTokyo Paired Ego-Video (PEV) Dataset contains 1226 paired egocentric videos recording dyadic human-human interactions [39]. It consists of 8 interaction categories and was recorded by 6 subjects. We split the data into train-test subsets based on the subject pairs as done in [39] and the mean accuracy of the three splits is reported.

NUS First-person Interaction Dataset contains 152 first-person videos and 133 third-person videos of both human-human and human-object interactions [25]. We evaluate our

Methods	PEV	NUS(first h-h)	NUS(first)	JPL
RMF[1]	-	-	-	86.0
Ryoo and Matthies[31]	-	-	-	89.6
Narayan *et al*. [25]	-	74.8	77.9	96.7
Yonetani *et al*. [39] (single POV)	60.4	-	-	75.0
convLSTM[35] (raw frames)	-	-	69.4	70.6
convLSTM[35] (difference of frames)	-	-	70.0	90.1
LRCN[6]	45.3	65.4	70.6	78.5
TRN[41]	49.3	66.7	74.7	84.2
Two-stream[33]	58.5	78.6	80.6	93.4
Our method	**64.2**	**80.2**	**81.8**	**98.4**
Yonetani *et al*. [39] (multiple POV)	69.2	-	-	-
Our method (multiple POV)	69.7	-	-	-

Table 1. State-of-the-art comparison (%) with existing methods. *NUS(first h-h)* denotes the first-person human-human interaction subset of NUS dataset and *NUS(first)* denotes the first-person subset. It is notable that only PEV dataset provides multiple POV videos so that no multiple POV result of other datasets is reported.

method on first-person human-human interaction subset to verify the effectiveness of our method. To further test our method in human-object interaction cases, we also evaluate on the first-person subset. Random train-test split scheme is adopted and the mean accuracy is reported.

JPL First-Person Interaction Dataset consists of 84 videos of humans interacting with a humanoid model with a camera mounted on its head [31]. It consists of 7 different interactions. We validate our method's effectiveness in this static observer setting and report the mean accuracy over random train-test splits.

5.2. Implementation Details

Network Details. In the motion module, we set 5 as the maximum displacement for the multiplicative patch comparisons. In the interaction module, we reduce the size of ego-feature and exo-feature to 256 and set 256 as the hidden size of LSTM blocks. 20 equidistant frames are sampled as input as done in [35].

Data Augmentation. We adopt several data augmentation techniques to ease overfitting due to the absence of large amount of training data. (1) Scale jittering [37]. We fix the size of sampled frames as 160×320 and randomly crop a region, then resize it to 128×256 as input. (2) Each video is horizontally flipped randomly. (3) We adjust the hue and saturation in HSV color space of each video randomly. (4) At every sampling of a video, we randomly translate the frame index to obtain various samples of the same video.

Training setup. The whole network is hard to converge if we train all the parameters together. Hence, we separate the training process into two stages. At the first stage, we initialize feature extraction module with ImageNet [29] pretrained parameters and train attention module, motion module and interaction module successively while freezing other parameters. At the second stage, the three modules are finetuned together in an end-to-end manner. We use Adam optimizer with initial learning rate 0.0001 to train our model using TensorFlow on Tesla M40, and decrease the learning rate when the loss saturates. To deal with overfitting, we further employ large-ratio dropout, high weight regularization and early stop strategies during training.

5.3. Comparison to the State-of-the-art Methods

We compare our method with state-of-the-arts and the results are shown in Table 1. The first part lists the methods using hand-crafted features. The second part presents some deep learning based action analysis methods (reimplemented by us except convLSTM). The third part reports the results of our method and the fourth part compares the performance using multiple POV videos on PEV dataset.

As shown, our method outperforms existing methods. Most previous methods directly learn interaction representations without relation modeling, while ours explicitly models the relations between the two interacting persons. The results show that relation modeling is useful for interaction recognition.

Among the compared deep learning methods, we obtain clear improvement over convLSTM[35], LRCN[6] and TRN[41], since they mainly capture the temporal changes of appearance features, but ours further explicitly captures motions and models the relation between the two interacting persons. Two-stream network [33] with the same backbone CNN as ours integrates both appearance and motion features but obtains inferior performance to ours, perhaps due to the lack of relation modeling.

On PEV dataset, Yonetani *et al*. [39] achieves 69.2% of accuracy with paired videos, certainly surpassing others using single POV video. We use our interactive LSTM to fuse the features from paired videos since there also exist some relations between the actions recorded by the paired videos.

Features	PEV	NUS(first h-h)
Ego-features	55.2	67.9
Exo-features	53.1	76.1
Concat(no relation)	60.8	77.9
Interaction with sym. blocks	62.7	78.1
Interaction with rel. branch	63.0	79.0
Interaction with both	64.2	80.2

Table 2. Recognition accuracy comparison (%) about interacion. *Concat(no relation)* means concatenation of ego-features and exo-features without any relation modeling. *Interaction with sym. blocks* means only symmetrical LSTM blocks are used. *Interaction with rel. branch* means only relation LSTM branch is used. *Interaction with both* means both components are used.

We achieve comparable result (69.7%) which further proves the relation modeling ability of our interactive LSTM.

In terms of inference time, our framework takes around 0.15 seconds per video with 20 sampled frames, which is still towards real time. TRN[41] takes 0.04 seconds per video but it has clear lower recognition performance than ours. Although Two-stream[33] obtains slightly inferior performance to ours, it takes 0.9 seconds per video since it spends much more time on extracting optical flows.

5.4. Further Analysis

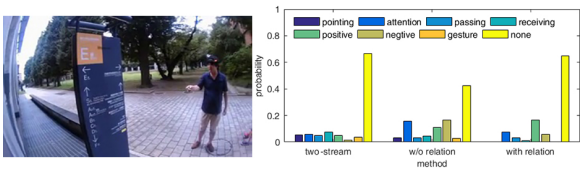
5.4.1 Study on Interaction Module.

Table 2 compares recognition performance about interaction. It shows that our interactive LSTM clearly improves the performance, since it models the relations and also drives feature learning of other modules. On different datasets, the relation modeling obtains different performance gains. We obtain clearer improvements on PEV dataset since it contains more samples dependent on relations. While in NUS(first h-h) dataset, most samples have weaker relations between the two interacting persons.

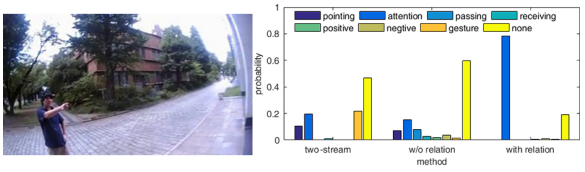
As discussed in Section 4, the main differences between the two interaction samples shown in Figure 6 might be the sequential orders and motion directions. We further compare the recognition results on them of different methods. It is observed that both two-stream[33] and simple concatenation cannot sufficiently model the two interactions. While with explicit relation modeling, the two interactions are correctly distinguished, which indicates that our interactive LSTM models the relations to distinguish confusing samples for better interaction recognition.

5.4.2 Study on Attention Module.

We compare recognition accuracy of different appearance features in Table 3. It is observed that local appearance features slightly improve the performance since it provides concrete descriptions of the interactor instead of general or



(a) Interaction category: none



(b) Interaction category: attention

Figure 6. Comparison of recognition results of two interaction samples. *w/o relation* means concatenation of two action features without any relation modelling is used for recognition. The bar graphs on the right present the probabilities of each category.

overall features, which are more related to the interaction. Furthermore, relation modeling performs better than concatenation since it enhances the features through symmetrical gating or updating and relation modeling.

Figure 7 visualizes some learned masks. (See more examples in supplementary material.) As shown, the attention module learns to localize the interactor with the JPPNet reference masks as supervision. With additional classification loss, it could localize some objects around the interactor and strongly related to the interactions such as the hat in the example, which leads to around 2% accuracy boost for local appearance features. This shows the advantage of using the designed attention module in our framework over using the JPPNet masks directly in this recognition task. In addition, with only the classification loss, our attention module fails to localize the interactor at all, indicating the necessity of reference masks for interactor localization.

The attention module is an indispensible part of our framework for individual action representation learning. It not only learns concrete appearance features, but also severs to separate the global and local motion for explicit motion features learning. Without attention module, our framework could only capture the global appearance and motion cues, and fails to model the relations between the camera wearer and the interactor, which leads to 9.0% and 12.3% accuracy degradation on PEV and NUS(first h-h) dataset, demonstrating the importance of attention module.

5.4.3 Study on Motion Module.

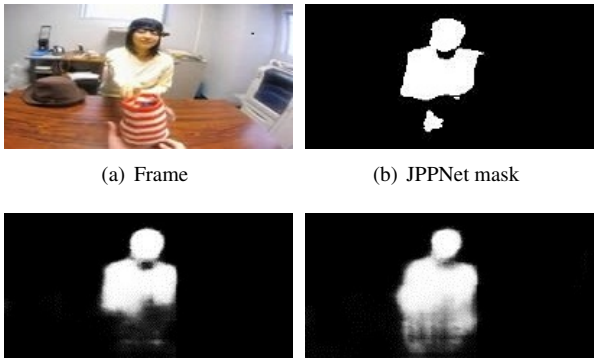
We show accuracy comparisons of different motion features in Table 4. It is seen that two-stream (flow) is a powerful method, but it is computational inefficient. Our method explicitly captures motions of the camera wearer and in-

Features	PEV	NUS(first h-h)
Two-stream[33](RGB)	40.7	63.8
Global appearance	40.7	63.8
Local appearance	43.2	65.1
Concat(no relation)	44.2	66.8
Interaction	45.9	68.2

Table 3. Recognition accuracy comparison (%) using appearance features. *Concat(no relation)* means simple concatenation of global and local appearance features. *Interaction* means relations modeling is used with global and local appearance features.

Features	PEV	NUS(first h-h)
Two-stream[33](flow)	54.0	73.2
Global motion	51.9	52.3
Local motion	51.0	69.6
Concat(no relation)	53.2	73.4
Interaction	56.6	75.0

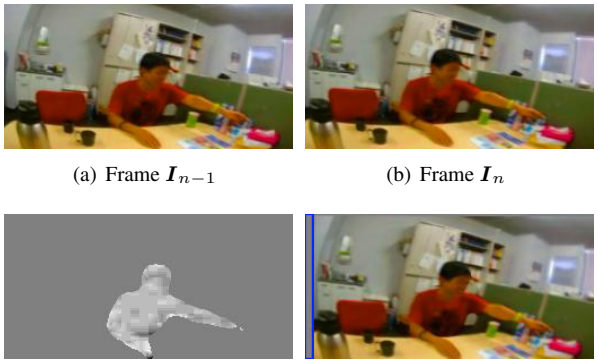
Table 4. Recognition accuracy comparison (%) using motion features. *Concat(no relation)* means simple concatenation of global and local motion features. *Interaction* means relations modeling is used with global and local motion features.



(a) Frame (b) JPPNet mask

(c) Mask (d) Mask

Figure 7. Example of learned mask with different supervision. (a) is original frame; (b) is the JPPNet mask; (c) is the learned mask trained with only human segmentation loss; (d) is the learned mask trained with human segmentation loss and classification loss.



(a) Frame $\boldsymbol{I}_{n-1}$ (b) Frame $\boldsymbol{I}_n$

(c) Local dense motion (d) Global motion

Figure 8. Illustration of global and local motion. (a) and (b) are two consecutive sampled frames. (c) Local dense motion shows the amplitudes of the horizontal motion vectors in the interactor mask, the amplitudes to the right are proportional to the brightness of the motion field. The motion vectors outside the interacor mask is discarded. (d) Global motion shows the slight head motion to the right, which reflects on the strip highlighted in blue.

teractor and reaches comparable results with two-stream (flow), which indicates the effectiveness of our motion module. Furthermore, our method could achieve higher accuracy with relation modeling. On different datasets, global motion and local motion contribute differently to recognition, probably because global motion is important to distinguish interactions such as positive and negtive response on PEV dataset, but such interactions highly relevant to global motion are not included in NUS(first h-h) dataset.

In Figure 8, we show the reconstructed frame and local dense motion field. (See more examples in supplementary material.) From the reconstructed frame, it is seen that the the slight head motion to the right is captured, which leaves a strip on the left highlighted in blue. The local dense motion field shows the motion of the interactor reaching out the hand towards the right. This example shows that the motion module could learn the global and local motion jointly.

Our motion module explicitly estimates global and local motions of the camera wearer and the interactor individually, which is important for relation modeling. Without the motion module, our method fails to capture motion information and can only use appearance features, which leads to 18.3% and 12.0% accuracy drop on PEV and NUS(first h-h) dataset, showing the necessity of motion modeling.

6. Conclusion

In this paper, we propose to learn individual action representations and model the relations of the camera wearer and the interactor for egocentric interaction recognition. We construct a dual relation modeling framework by developing a novel interactive LSTM to explicitly model the relations. In addition, an attention module and a motion module are designed to jointly model the individual actions of the two persons for helping modeling the relations. Our dual relation modeling framework shows promising results in the experiments. In the future, we would extend our method to handle more complex scenarios such as multi-person interactions, which are not considered in this paper.

Acknowledgement

This work was supported partially by the National Key Research and Development Program of China (2018YFB1004903), NSFC(61522115), and Guangdong Province Science and Technology Innovation Leading Talents (2016TX03X157).

References

[1] Girmaw Abebe, Andrea Cavallaro, and Xavier Parra. Robust multi-dimensional motion features for first-person vision activity recognition. *Computer Vision and Image Understanding*, 149:229–248, 2016.

[2] Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *Advances in neural information processing systems*, pages 5074–5082. 2016.

[3] Stefano Alletto, Giuseppe Serra, Simone Calderara, and Rita Cucchiara. Understanding social relationships in egocentric vision. *Pattern Recognition*, 48(12):4082–4096, 2015.

[4] Stefano Alletto, Giuseppe Serra, Simone Calderara, Francesco Solera, and Rita Cucchiara. From ego to nos-vision: Detecting social relationships in first-person views. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 580–585, 2014.

[5] Sven Bambach, Stefan Lee, David J Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1949–1957, 2015.

[6] J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):677–691, 2017.

[7] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766, 2015.

[8] Wenbin Du, Yali Wang, and Yu Qiao. Rpan: An end-to-end recurrent pose-attention network for action recognition in videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3745–3754, 2017.

[9] Wenbin Du, Yali Wang, and Yu Qiao. Recurrent spatial-temporal attention network for action recognition in videos. *IEEE Transactions on Image Processing*, 27(3):1347–1360, 2018.

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

[11] Alircza Fathi, Jessica K Hodgins, and James M Rehg. Social interactions: A first-person perspective. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1226–1233, 2012.

[12] Alireza Fathi, Yin Li, and James M Rehg. Learning to recognize daily actions using gaze. In *The European Conference on Computer Vision (ECCV)*, pages 314–327, 2012.

[13] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

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

[16] Reza Kahani, Alireza Talebpour, and Ahmad Mahmoudi-Aznaveh. A correlation based feature representation for first-person activity recognition. *arXiv preprint arXiv:1711.05523*, 2017.

[17] 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.

[18] J. Lee and M. S. Ryoo. Learning robot activities from first-person human videos using convolutional future regression. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1497–1504, 2017.

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

[20] 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.

[21] X. Liang, K. Gong, X. Shen, and L. Lin. Look into person: Joint body parsing amp; pose estimation network and a new benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4):871–885, 2019.

[22] 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)*, pages 1894–1903, 2016.

[23] Yang Mi, Kang Zheng, and Song Wang. Recognizing actions in wearable-camera videos by training classifiers on fixed-camera videos. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 169–177, 2018.

[24] T. P. Moreira, D. Menotti, and H. Pedrini. First-person action recognition through visual rhythm texture description. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2627–2631, 2017.

[25] Sanath Narayan, Mohan S Kankanhalli, and Kalpathi R Ramakrishnan. Action and interaction recognition in first-person videos. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 526–532, 2014.

[26] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1520–1528, 2015.

[27] 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*, 29(3):773–786, 2019.

[28] 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)*, pages 2847–2854, 2012.

[29] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy,

Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[30] MS Ryoo, Thomas J Fuchs, Lu Xia, Jake K Aggarwal, and Larry Matthies. Robot-centric activity prediction from first-person videos: What will they do to me? In *ACM/IEEE International Conference on Human-Robot Interaction*, pages 295–302, 2015.

[31] Michael S Ryoo and Larry Matthies. First-person activity recognition: What are they doing to me? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2730–2737, 2013.

[32] M. S. Ryoo, B. Rothrock, and L. Matthies. Pooled motion features for first-person videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 896–904, 2015.

[33] 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.

[34] 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)*, pages 2620–2628, 2016.

[35] Swathikiran Sudhakaran and Oswald Lanz. Convolutional long short-term memory networks for recognizing first person interactions. In *IEEE International Conference on Computer Vision Workshops*, pages 2339–2346, 2017.

[36] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[37] 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 *The European Conference on Computer Vision (ECCV)*, pages 20–36, 2016.

[38] Lu Xia, Ilaria Gori, Jake K Aggarwal, and Michael S Ryoo. Robot-centric activity recognition from first-person rgb-d videos. In *IEEE Winter Conference on Applications of Computer Vision*, pages 357–364, 2015.

[39] Ryo Yonetani, Kris M. Kitani, and Yoichi Sato. Recognizing micro-actions and reactions from paired egocentric videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2629–2638, 2016.

[40] Hasan FM Zaki, Faisal Shafait, and Ajmal Mian. Modeling sub-event dynamics in first-person action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1619–1628, 2017.

[41] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *The European Conference on Computer Vision (ECCV)*, 2018.

[42] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6612–6619, 2017.

[43] 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)*, pages 1904–1913, 2016.