Recurrent Modeling of Interaction Context for Collective Activity Recognition

Minsi Wang, Bingbing Ni, Xiaokang Yang
Shanghai Jiao Tong University
mswang1994@gmail.com, {nibingbing,xkyang}@sjtu.edu.cn

Abstract

Modeling of high order interactional context, e.g.*, group interaction, lies in the central of collective/group activity recognition. However, most of the previous activity recognition methods do not offer a flexible and scalable scheme to handle the high order context modeling problem. To explicitly address this fundamental bottleneck, we propose a recurrent interactional context modeling scheme based on LSTM network. By utilizing the information propagation/aggregation capability of LSTM, the proposed scheme unifies the interactional feature modeling process for single person dynamics, intra-group (*e.g.*, persons within a group) and inter-group (*e.g.*, group to group) interactions. The proposed high order context modeling scheme produces more discriminative/descriptive interactional features. It is very flexible to handle a varying number of input instances (*e.g.*, different number of persons in a group or different number of groups) and linearly scalable to high order context modeling problem. Extensive experiments on two benchmark collective/group activity datasets demonstrate the effectiveness of the proposed method.*

1. Introduction

Analysis of collective activity groups provides useful information for several real-world applications including social role understanding and social event prediction. The main challenge of collective activity recognition is modeling of interactional context information among persons. This is because that the number of persons involved in an interaction is always varying. Moreover, in most cases a collective activity is associated with several sub groups of interactions, and how to model the group to group interaction is even more challenging.

Previous methods for activity recognition mainly focuses on modeling unary features, *e.g.*, single person appearance or dynamics information [21, 26] and person to person interaction (*e.g.*, pairwise features) [22]. However, these contextual information modeling schemes are not sufficient for collective activity recognition. It is because that in collective activity, different activity categories might share the same type of unary or pairwise features (*e.g.*, "standing alone" in the cases of queueing or discussion, "facing to same direction" in the case of *walking* and *crossing*). In other words, besides modeling the intra-group interaction (*e.g.*, interaction among the persons within a group), how to effectively describe the group to group interaction is of more importance. Low order contextual features do not provide sufficient cues to recognize these activities. To address this fundamental problem, most previous methods attempt to encode the high order relationship among persons in the scene by inferring the latent graphical structures [9, 8]. However, applying these approaches for collective activity recognition is infeasible because these methods often requires high computational cost in the case of tree-structured model during inference and learning. Moreover, it is very difficult to generalize methods based on graphical models to handle higher order interactional context. Ni *et al*. [24] proposed a causality analysis framework to encode unary, pairwise and group interaction features. However, this method is only capable of modeling human trajectory level information, which is insufficient to recognize finer-grained actions, *e.g.*, those can only be recognized by human appearance or local body part dynamics.

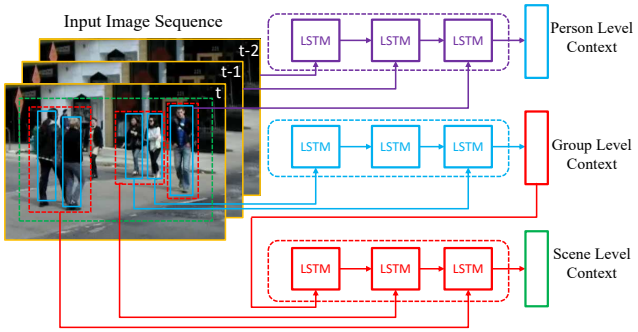


Figure 1. The overview of proposed framework. A hierarchical recurrent interactional context modeling framework is proposed to model intra-group and inter-group interaction context.

A fundamental problem becomes: how to systematically encode the high order human interactional context, i.e, the

target method should be feasible for representing contextual features among arbitrary number of interacting persons or groups and the computational complexity scales well with the context order. To this end, we propose a **recurrent interactional context encoding** scheme based on Long-Short Term Memory (LSTM) [16].

More specifically, we propose a hierarchical recurrent interactional context encoding framework to handle three level interactions, namely single human dynamics, within group human interaction and group to group interaction. First, spatio-temporal graph partition is performed to form human interactional subgroups. Besides, we propose a unified network architecture to model single person dynamics, intra-group (persons within a interactional subgroup) and inter-group (group to group) contextual features, based on the sub-action encoder and long short-term memory (LSTM) network. Moreover, we also propose to input unary features encoded by sub-actions (*e.g.*, *move* or *pose*) within a spatial-temporal interaction group into our hierarchical LSTM context encoding network. Here are the advantages of this scheme. First, encoding each person dynamics by sub-actions is sufficient enough to distinguish different actions. Second, using the recurrent contextual information accumulation/modeling scheme, modeling of different order of contextual information is thus unified, as LSTM nodes share parameters and therefore increasing the order can be simply handled by adding another LSTM node. Third, the model complexity of proposed scheme is linearly scalable with respect to the context order. Extensive experimental results on two benchmark collective/group activity datasets well demonstrate the discriminative capability, flexibility of modeling high order interactional context and robustness to noisy human detections, by comparing to the state-of-the-art group activity recognition methods.

The rest of the paper is organized as follows. We review some related works in Section 2. In Section 3, we introduce the proposed recurrent interactional context encoding framework and the implementation details. Extensive experimental results and discussion are presented in Section 4. Section 5 gives the conclusions.

2. Related Work

Collective/Group Activity Recognition. Many pervious works have been done on collective activity recognition focusing on contextual learning [6, 7], where the spatial distribution of atomic activities are applied to describe group activities. Amer *et al.*[2] detected the video parts where the collective activities occur and made use of these local visual cues in the detected parts for recognition. Lan *et al.* [22] proposed an adaptive latent structure learning that represents hierarchical relationships ranging from lower person-level information to higher group-level interactions. In [21] and [26] the idea of social roles, the expected behaviour of an individual person in the context of group, is exploited in fully supervised and weakly supervised frameworks respectively. Choi and Savarese [5] have unified tracking multiple people, recognizing individual actions, interactions and collective activities in a joint framework. In [7], a random forest structure is used to sample discriminative spatio-temporal regions from input video fed to 3D Markov random field to localize collective activities in a scene. Shu *et al.* [30] detect group activities from aerial video using an AND-OR graph formalism. Recently, a probabilistic structured kernel method constructed based on a multi-instance cardinality model is introduced in [15]. Furthermore, Deng *et al.* [9] introduced a neural network-based hierarchical graphical model that predict group activity simultaneously. In [17], a LSTM based hierarchical deep temporal model is proposed to model temporal dynamics for group activity recognition.

Recurrent Neural Networks and LSTM. Recurrent neural networks especially the long-short term memory models [16] have achieved great success in a large variety of applications including temporal modeling such as natural language processing [33, 34] and speech recognition [14, 13], and non-temporal modeling such as image caption generation [19, 37]. Several works have been proposed to model action image sequences using RNN/LSTM models. Veeriah *et al.* [36] proposed a differential gating scheme for the LSTM neural network, which emphasizes on the change in information gain caused by the salient motions between successive frames. Donahue *et al.* [10] developed a novel recurrent convolutional architecture for large-scale visual learning. They applied this model on several tasks including benchmark video recognition, image description, and video narration. Karpathy *et al.* [19] described a multi-modal RNN architecture to generate image descriptions. Wu *et al.* [38] extracted spatial and the short-term motion features by two CNNs to further model longer-term temporal clues. The two types of CNN-based features are further combined in a regularized feature fusion network for video event classification.

Recently, LSTM network is also applied in fine grained action detection [25], human trajectory prediction [1], and in object recognition in the context of recurrent visual attention [3] [23] [29]. For collective activity recognition, Ibrahim and Muralidharan [17] introduce a hierarchical structured model, which incorporates a deep LSTM framework to recognize individual actions and group activities. They leverage LSTM-based temporal modelling to learn discriminative information from time varying sports activity data. However, person pooling is not able to model group to group context. To solve this problem, in our works, a hierarchical recurrent interactional context encoding framework is proposed to model intra-group and inter-group interaction context.

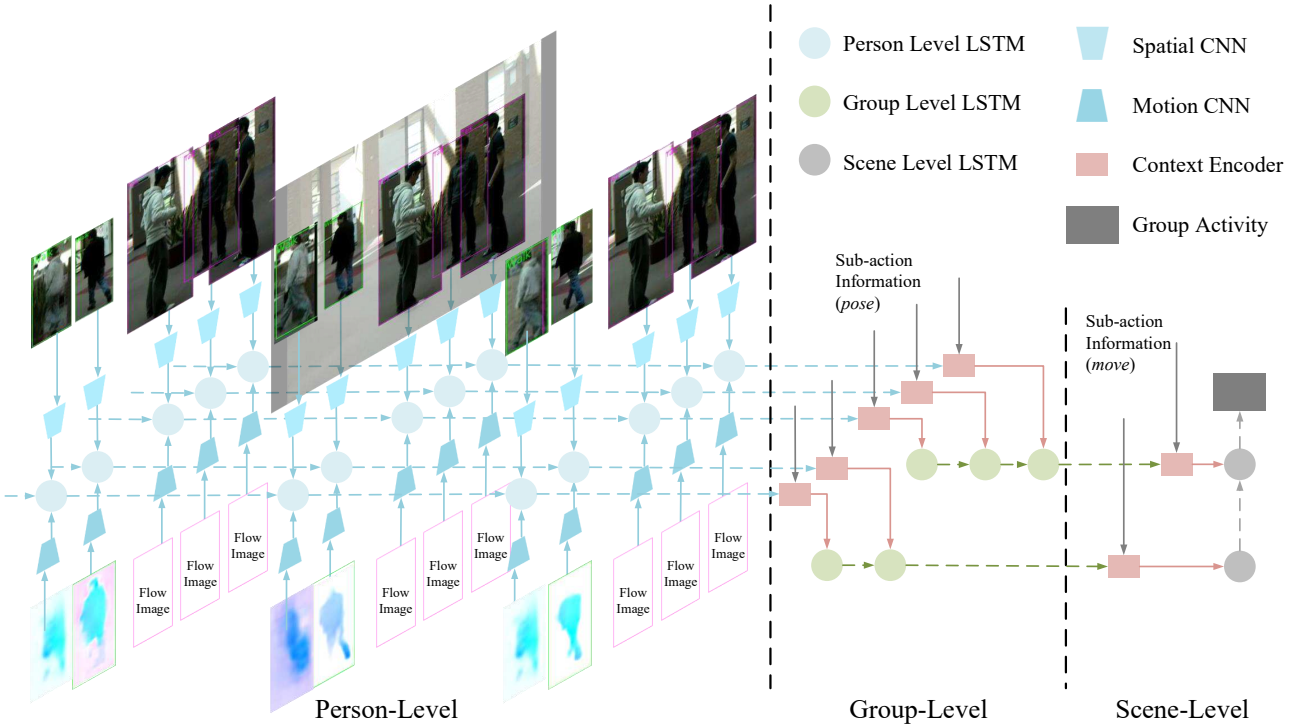


Figure 2. The hierarchical recurrent interactional context encoding framework for collective/group activity recognition. Given tracklets of N^i person, we feed each tracklet into spatial CNN and motion CNN respectively and concatenate their outputs, followed by a person level LSTM network to represent person dynamics. Then, we utilize context encoder to model group level and scene level interaction context. In the end, the encoding results are fed into LSTM network to identify the whole activity.

3. Methodology

In contrast to traditional activity recognition, collective activity has its unique natures. In particular, the interaction between human groups plays more important role in collectivity (*e.g.*, two groups stand face to face in the activity *discussion*). To expand the difference between group actions and to better represent them, we deploy contextual binary encoder in our hierarchical group activity recognition scheme, which encodes the sub-action (*e.g.*, *move* and *pose*) information of people into person dynamics to enrich person level features and make person action unique.

To model group-to-group interaction, a pipeline is proposed as follows (which is illustrated in Figure.2). Firstly, we perform human detection and tracking to generate human tracklets (*e.g.*, a sequence of tracked human bounding boxes). Then, we apply clustering/segmentation method to partition all human tracklets into spatio-temporal consistent groups. After that, we train the proposed hierarchical recurrent context encoding network to learn interactional context features for 1) single human dynamics, 2) intra-group human interactions, and 3) inter-group interactions.

3.1. Interaction Volume Generation

Generate Human Tracklet. For fair comparison, the input to our method is a set of tracklets of the people in a scene provided by Choi *et al*. [5, 6].

Generate Human Groups. The key step to recognize collective/group activities is to model the interactions between human groups as well as the interactions among each group, therefore we must partition all the single human tracklets in the video into human groups in prior to further processing. We perform tracklets grouping/partition based on the graph partition algorithm used as in [27]. The adjacency graph is constructed according to the relative spatial distance and velocity between tracklets.

3.2. Context Encoding of Group Interaction

The goal of this work is to model group interaction in collective activities. As discussed above, the input to our group interaction modeling framework is a set of human tracklets in the video as well as groupings of these tracklets. To this end, we build a hierarchical scheme which models single person appearance dynamics, encodes sub-action information to obtain within group human interaction and group to group interaction in a bottom-up manner. For each level of interaction modeling, sub-action encoder and RNN (*e.g.*, LSTM) is utilized to aggregate a varying number of entity level features as well as the relationship between entities to a unified context feature representation, *i.e.*, an entity is corresponding to a person for within group context modeling, or a human group for group to group context modeling. Details are given as follows.

3.2.1 Single Person Interaction Context

Single person interaction context includes two cues: 1) change of appearance of a person over time and 2) temporal dynamics of a person's action, which provide important cues for recognizing collective/group activity. For instance, to distinguish between walking and queuing, whether the person is standing still or moving is discriminative.

More specifically, given the tracklet (tracked spatio-temporal volume) of a person (I_i for tracklet in original image and I_i^f for tracklet in corresponding flow image), we employ long short-term memory (LSTM) models to encode temporal evolution of an individual person. Motivated by the success of deep convolutional neural network features (DCNN) in representing image patch/region level visual characteristics and optical flow algorithm in representing motion of objects, in this work, similar to [31, 12] we use Flownet [11] to generate flow images for each frame and extract DCNN features from each human bounding box along the tracklet (both original images and flow images), which serve as the input sequence to the LSTM model.

We denoted by $X = \{x_1, x_2, \cdots, x_T\}$ the sequence of input features, *i.e.*, x_t the feature vector fused by concatenating original and flow features input to the t-th LSTM node. The corresponding state and output of each LSTM node is denoted by h_t and o_t, respectively. Each LSTM node includes three gates, (*i.e.*, input gate i, output gate o and forget gate f) as well as a memory cell. At each time step t, given the input x_t and the previous hidden state h_{t-1}, LSTM updates as follows:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + V_f c_{t-1} + b_f) \quad (2)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (3)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_t + b_o) \quad (4)$$

$$h_t = o_t \odot tanh(c_t) \quad (5)$$

where σ is the sigmoid function and $\odot$ denotes the element-wise multiplication operator. W_*, U_* and V_* are the weight matrices, and b_* are the bias vectors. The memory cell c_t is a weighted sum of the previous memory cell c_{t-1} and a function of the current input. The weights are the activations of forget gate and input gate respectively. On the one hand, the hidden state h_t could be used to represent the specific atomic action the person is performing at time t, e.g., walking or standing still. On the other hand, h_t also contains the aggregated action information of that person from the first time stamp to t, *e.g.*, person dynamics.

3.2.2 Intra-Group and Inter-Group Interaction Context

As mentioned above, person in the scene are partitioned into several groups according to their spatio-temporal proximity. It is then important to model the interaction within each group, *i.e.*, person to person interaction. Most previous methods on interaction modeling are usually based on the pairwise features, *e.g.*, relative distance or relative velocity between persons, and these methods are difficult to be generalized to cope with higher order interactional context, *i.e.*, when the persons in the group is more than two.

To address this issue, we propose a LSTM based context encoding framework to model interactional context. Namely, we first encode the person level features based on their sub-actions (*e.g.*, *move* and *pose*), then order the person level features by x or y coordinate of the center of person in image and input them into another LSTM network. The aggregated output of this LSTM network serve as the intra-group person to person interactional context. In other words, person level dynamics are enriched by encoder and collected over all people in the group, so that it can be used to describe the group interactional activity within the group. Note that this scheme is flexible to encode the contextual information among arbitrary number of persons in the group. We will first introduce **context encoding** and then illustrate its application in inter-group and intra-group interaction context modeling.

Context Encoding. The single person level modeling has recognized overall person dynamics and to model the intra-group and inter-group interaction context, more information in detail is needed to explore the pattern of group action. Some graphical structures [5, 22, 26] have been discovered to model pairwise interaction context, however they can not represent whole interaction context sufficiently and efficiently. Here, context encoding is proposed to model whole interaction context among all people in image and it will scales well with increasing number of entities involved in the context (*i.e.*, context order).

Inspired by [6], we use spatial-temporal information to encode context. In Figure.3, given sub-action information for each person, our context encoder aims to encode sub-action information into person dynamics to enrich person level features. Two types of sub-actions (*e.g.*, *move* and *pose*) are deployed in encoder, which is enough to describe the action of individuals in our experiments (Section 4).

According to traditional binary code, there are usually $\{0, 1\}$ after encoding. But for neural network, input $\{0\}$ usually means no input. Thus, we define $\{-1, 0, 1\}$ to represent the code where $\{0\}$ means the element makes no sense. For encoding sub-action *move*, there are three elements $\{-1, 0, 1\}$ to be represented, where $\{-1, 0, +1\}$ denotes move left, stand still and move right respectively (dotted lines in Fig.3). We encode the sub-action *move* by calculating the motion in x coordinate as Eq. (6).

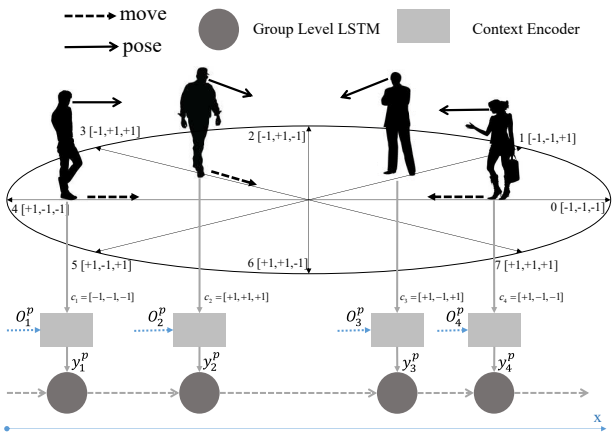


Figure 3. Context Encoder. The motion of anchor person is divided into two direction (*e.g.*, left and right). The pose of the anchor person is divided into eight orientation encoded by the number around circle. The inputs of context encoder are sub-action information c_i^k and person dynamics O_i^p and its output is y_i^p in Eq. (7). y_i^p is fed into group LSTM following x coordinate order.

$$C(\triangle x_{I_b}) = \begin{cases} -1, & \triangle x_{I_b} < \mu \\ 0, & |\triangle x_{I_b}| \leqslant \mu \\ +1, & \triangle x_{I_b} > \mu \end{cases} \quad (6)$$

where $\triangle x_{I_b} = \left(x_{I_b}^t - x_{I_b}^{t-1}\right)$ and $x_{I_b}^t$ denotes the x coordinate of the center of bounding box I_b of corresponding person in time stamp t. μ indicates the threshold for determining whether person moves (*e.g.*, if motion of person is less than μ pixels, the person is treated as stand still). Moreover, to represent the sub-action of eight types of action *pose* (*e.g.*, eight directions people face to) annotated in dataset, we utilize 3-bit code (range from 0 to 7) to represent eight orientation for *pose* and the way to encode *pose* is shown as solid lines in Figure.3.

Also, to better represent person's character in group, sub-action information (*e.g.*, *move* and *pose*) is encoded into the feature of person dynamics according to Eq. (7),

$$y_i^p = [c_i^1 \times O_i^p, c_i^2 \times O_i^p, ..., c_i^k \times O_i^p] \quad (7)$$

where c_i^k denotes k th bit of code for i th person in group, O_i^p is the output of person level LSTM (p indicates person level in O_i^p) for i th person and $[\,]$ denotes vector concatenation in Eq. (7). (*e.g.*, assume the dimension of O_i^p is 1×1024, $[c_i^k]$ is 4 bits code then the dimension of y_i^p is 1×4096). We use binary code because of its nature that every bit has two types of opposite value corresponding to opposite direction in sub-actions (*e.g.*, *move* or *pose*). In our paper, it is strongly recommended to perform the sub-action augmentation for model robustness (Section 3.3).

Inter-group and Intra-group Interaction Context Modeling. Next, for within group interaction modeling, after performing context encoding, encoded features y_i^p can be directly fed into intra-Group or inter-Group LSTM network for training (Group-Level or Scene-Level in Figure.2). Given the annotation of frame, LSTM can learn the coding rule and is able to decode the code at prediction phase.

Significantly, one problem is that recursive network including LSTM only accepts *ordered* input sequence, *i.e.*, time series. However, in our case the set of person level features are orderless. Therefore, a fundamental processing step is to perform some *ordering/alignment* scheme to facilitate the subsequent processing. For such a purpose, we use spatial cues to order the encoded person level features within a group to input to the LSTM network. More specifically, we order the person level features by the x or y coordinates of the respective tracklets and form two LSTM input sequences (x coordinate is better in our works). For each direction, we obtain a spatially aggregated interaction context representation which indicates the person to person interaction along that direction (*e.g.*, ordered by x coordinate from left to right in Figure. 3).

The modeling of group to group (inter-group) interactional context is similar with intra-group context modeling. Namely, group level representations are first encoded by common sub-action information (usually different from within group), ordered by the x or y coordinates of the geometric centers of each group and input to another LSTM network and the aggregated network output serves as the scene level (group to group) interaction representation.

3.3. Implementation Details

We trained our model in three steps (person-level, group-level, scene-level). In addition, there is context encoding step when modeling intra-group and inter-group context. The network structures and parameters of our proposed hierarchical model are defined as follows:

1. Level 1 Network (Person Context). We extract D-CNN features based on models pre-trained on ImageNet [28]. Two CNN networks are applied, and the spatial CNN (AlexNet [20]) is for original images and the motion CNN (GoogleNet [35]) is for flow images. Typically, the motion CNN is not needed to train when performing person level training. More specifically, a single layer LSTM network is placed after concatenation layer. Therefore the dimensionality of the LSTM cell input is $4096 + 1024 = 5120$. Each LSTM layer contains 1024 hidden units. The number of output units is set as the number of class (*e.g.*, action label or scene label).

2. Context Encoding. The person level features are transformed following Eq. (6) (7) and the encoded features are the input of next network after being ordered by the x coordinate.

3. Level 2 Network (Group Context). The input to this LSTM network is the output of the person level context network after context encoding. The outputs of spatial CNN, motion CNN and person level LSTM are concatenated, therefore the dimensionality of the input vector to each LSTM cell is $4096 + 1024 + 1024 = 6144$ if context encoding step do not change its dimension. Single layer LSTM network is adopted, and each LSTM layer contains 1024 hidden units.

4. Level 3 Network (Scene Context). The input to this LSTM network is the output of the group level context network after context encoding, and the structure is similar as group level network. Differences between group level and scene level network are the dimensionality of input that depends on encoding results.

The training process is performed on Caffe [18]. All the input patches of images are resized to 227×227 pixels and subtracted by the image mean. Our training procedure follows a bottom up manner. Namely, we firstly train the person level context network. The output of the level 1 network serves as the input to Level 2 network after context encoding to train the group context network. Finally, the output of Level 2 network is used to train Level 3 scene context network after context encoding (Figure. 2). For Level 1 network, the input DCNN features are extracted by the ImageNet pre-trained model.

To train all networks, the learning rate for person level network is fixed value 0.00001. The original learning rate for LSTM network is 0.0001, and the learning rate is decreased to $\frac{1}{2}$ of the original value after every two epochs. All LSTM networks are trained/tested using the implementation of [10]. To represent person level context for a video sample, we input all the person tracklets into Level 1 network and choose the last one of outputs as feature vector. Similarly, to represent group level context for a video sample, the last one of level 2 network outputs is chosen as the input of next level network.

Data Augmentation for Sub-action. Due to the lack of training data and the difference between training and testing data, our model cannot encode all the direction of *move* and orientation of *pose*. In order to increase model robustness, a novel data augmentation method is deployed to avoid the weakness of context encoding. We perform data augmentation not only augment the diversity of input image but also augment the diversity of composition of binary codes. The data augmentation method for sub-action is shown in Figure. 4. Assume that there are two people in group and the corresponding pose labels are $\{0, 5\}$ (blue lines in Figure. 4). For every person in image, we do the augmentation by rotating $\frac{360^\circ}{2} = 180^\circ$ one time for *move* or $\frac{360^\circ}{8} = 45^\circ$ seven times for *pose* (red lines in Figure. 4). Then binary codes are changed, but interaction context between them is kept (group activity is not changed). We aim to cover all situation by doing proposed sub-action data augmentation.

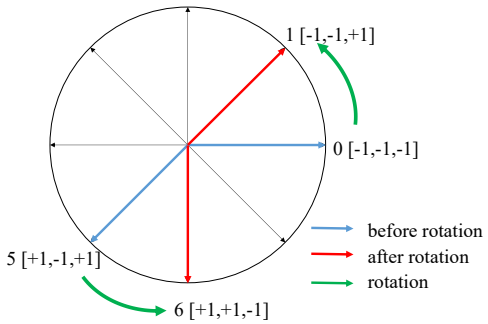


Figure 4. Data augmentation method for sub-action.

4. Experiments

We perform extensive experiments on the Collective Activity Dataset [6] and Choi's new Dataset [5] to validate the ability of learning context information, also compare our results with the state-of-the-art methods. And in depth discussions are also provided.

Due to the fact that the popular dataset introduced in [6] and [5] lacks sufficient diversity of background and large enough training data, sometimes what happens in image can be inferred by classifying the background of that image. In order to avoid the effects of background and focus on the analysis of interaction, proposed method ignores the information of background and does not use it in any steps.

4.1. Collective Activity Dataset

The Collective Activity Dataset has been widely used for evaluating group activity recognition performance. This dataset contains 44 video clips acquired by using low resolution handheld cameras. And there are eight person-level pose labels, five person level action labels, and five group-level activities in this dataset. A scene is simply assigned with what the majority of people are doing. We follow the train/test splits as suggested in [22], use the tracklet data provided in [5] and only use five group-level activities for training. Context encoder is not applied.

The Collective Activity Dataset consists of collective activities, *Crossing*, *Standing*, *Queuing*, *Walking* and *Talking*. According to [6], the *Walking* class is ill-defined as it is more like a single person activity than a collective one. Additionally, the only difference between class *Walking* and *Crossing* is the relation between person and street. Therefore, we merge class *Walking* and *Crossing* as class *Moving* and report the Mean Per Class Accuracy (MPCA) of *Walking* and *Crossing* as the accuracy of *Moving*. Due to the imbalanced test set, we report MPCA.

For person level network, proposed method using spatial

Table 1. Results (%) on Collective Activity Dataset [6]. The results of class *Walking* and *Crossing* are merged as *Moving*. *Pooling* denotes pooling-SVM method. Mean Per Class Accuracy (MPCA) are shown for comparsion.

Class	[22]	[5]	[17]	[15]	*Pooling*	Ours
Moving	92	90.0	**95.9**	87	94.2	94.4
Waiting	69	**82.9**	66.4	75	50.3	63.6
Queuing	76	95.4	96.8	92	100.0	**100.0**
Talking	99	94.9	**99.5**	99	99.5	**99.5**
MPCA	84	**90.8**	89.7	88.3	85.8	89.4

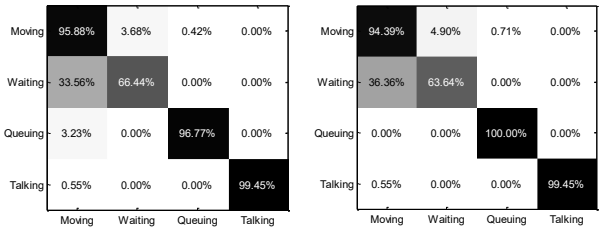
and motion CNN to recognize action is inspired by [31, 12] and it is most similar to the work [17]. Thus, we implement the model of [17] by performing a pooling-SVM structure with motion feature and no person level annotation (Table 1, *Pooling*). The results of our method are shown in Table 1 and compared with the following methods (1) Lan *et al*. [22], (2) Choi *et al*. [5], (3) Ibrahim *et al*. [17], (4) Hajimirsadeghi *et al*. [15]. Note that results of others are calculated from corresponding original confusion matrix in [22, 5, 17, 15].

As shown in Table 1 and Figure 5, the performance of our method is on the same level with the state-of-the-art methods. All of our results are comparable to baseline [17]. The difference between [17] and our model is that we do not use person action label but utilize optical flow information for training and testing. And it is clear that our model has improved performance compared to pooling-SVM method (*i.e*., the method [17] with motion feature and no person level annotation). It demonstrates that LSTM can aggregate context information and its feature aggregation ability can be used in feature pooling. The result of pool-SVM has a lower accuracy than our method. It is partly because pooling method can not well distinguish the class *Waiting* and *Queuing* which have nearly the same motion features. Besides, compared to pooling-SVM and our model with others, it shows that classification performance for the action class *Waiting* is unsatisfactory. Note that we do not use person-level pose labels and person level action labels, and class *Waiting* always occurs with class *Crossing* and *Walking* at the same time, which may be a factor in confused prediction.

From this experiment, we validate the ability of learning context information for LSTM and show a novel method of recognizing group activity without person level annotation by using LSTM to aggregate features.

4.2. Choi's New Dataset

The Choi's New Dataset [5] is composed of 32 video clips with 6 collective activities: *gathering*, *talking*, *dismissal*, *walking together*, *chasing* and *queueing*. There are 9 interaction labels, 3 atomic action labels, 8 pose labels and 6 group-level activities. We use all labels for training except 9 interaction labels. The atomic actions are labelled



(a) Baseline (b) Ours

Figure 5. Confusion matrix on Collective Activity Dataset [6] (regard $Walking$ and $Crossing$ as the same class $Moving$). (a) Confusion matrix of baseline [17]; (b) Confusion matrix of our method.

as *walking*, *standing still*, and *running*. The whole sets are divided into 3 subsets[1] and we run 3-fold training and testing as suggested in [5]. In experiments, we use the tracklets provided on the website of the authors of [6].

We employ hierarchical recurrent interactional context encoding framework to recognize group activity on this dataset. Firstly, person level network is applied to determine atomic action of each person in image. We encode each feature according to group information generated by the method in Section 3.1. In the end, encoded features are fed into group level or scene level network to determine final class. Note that we use various thresholds for encoding sub-action *move* (*e.g*., $Th0$, $Th1$ and $Th2$ denote models that threshold μ equals to $0, 1, 2$ respectively in Table 2), which is the boundary we define *move left or right* and *stand still* (*e.g*., if motion of person is less than 2 pixels, the person is treated as *stand still* in $Th2$ model).

We also compare our activity recognition results with the state-of-the-art methods. The methods for comparison includes: 1) Chang *et al*. [4], 2) Choi *et al*. [6] and 3) Choi *et al*. [5]. We implement the method [17] with motion features (*i.e*. optical flow images) on this dataset for comparison. As shown in Table 2, our method achieves a remarkable breakthrough in recognizing group activity in Choi's new dataset [5]. Our model namely $Th0$ further improves the Multi-Class Accuracy (MCA) performance to 89.4% and $Th1$ achieves best MPCA, 85.2%. This demonstrates that our contextual modeling scheme is effective.

Actually, there are atomic labels (move, stand still and running) instead of activity labels for each person in second dataset. So in Table 2, the method [17] doesn't work well without context encoder. Compared with its results, our method improves the performance by a large margin in both MCA and MPCA.

The confusion matrix of our method is also illustrated in Figure. 6. We note that the classification performance for the class *dismissal* is relative low, while the classifica-

[1] test set 1: [1, 2, 7, 12, 13, 19, 20, 21, 26, 27, 30];
test set 2: [3, 5, 10, 11, 15, 16, 17, 18, 24, 25, 31];
test set 3: [4, 6, 8, 9, 14, 22, 23, 28, 29, 32];

Table 2. Results (%) on Choi's New Dataset [5]. Both multi-class accuracy (MCA) and MPCA are shown because of class size imbalance. $Th0$ denotes the threshold μ is zeros while $Th1$ and $Th2$ denote threshold μ is equal to 1 and 2 pixels respectively.

Class	[4]	[6]	[5]	[17]	$Th0$	$Th1$	$Th2$
Gathering	59.9	50.0	43.5	30.7	71.9	71.9	72.8
Talking	97.0	72.7	82.2	91.4	95.9	95.9	95.9
Dismissal	90.5	49.2	77.0	31.6	68.4	73.7	74.7
Walking	94.3	83.2	87.4	82.4	86.6	85.2	78.9
Chasing	53.9	95.2	91.9	82.3	89.2	89.2	89.2
Queuing	86.3	95.9	93.4	69.6	95.5	95.5	95.5
MCA	-	77.4	83.0	78.1	**89.4**	89.2	87.3
MPCA	80.3	74.3	79.2	64.7	84.6	**85.2**	84.5

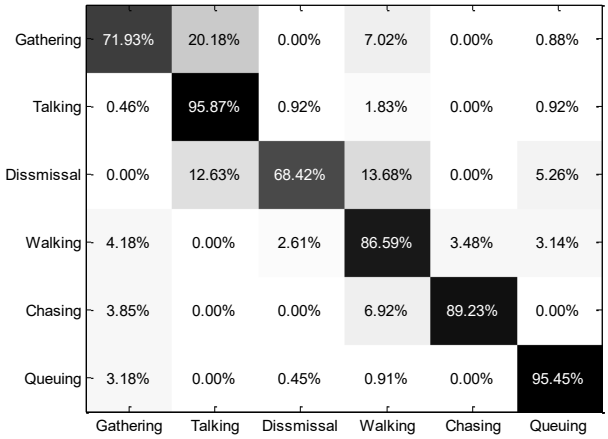


Figure 6. Confusion matrix on Choi's New Dataset [5] obtained by using our hierarchical recurrent interactional context encoding model.

tion accuracy for the class *gathering* is higher. On the one hand, this is due to the reason that context encoder cannot distinguish "moving into image plane" from "moving out of image plane". On the other hand, action *gathering*, *talking* and *dismissal* are temporal continuous process in the same video, as a result, it is hard to determine which class it belongs to in the transition from *gathering* into *talking* and the transition from *talking* into *dismissal*.

4.3. Discussion

Above all, LSTM based recurrent interactional context encoding scheme for group activity recognition is feasible in feature aggregation and it is predictable that it will success in processing large database in the future. In future, we attempt to boost performance by applying VGGnet [32] in our model.

Besides, proposed model works well in modelling group action which consists of various sub-actions. For view variance, ordering persons is an indirect way to model relative spatial information among each person and it does not intend to get absolute ordering. Thus, in general, it does not matter whatever the view angle changes. What the people do in corresponding ordering is important (See Figure. 7).

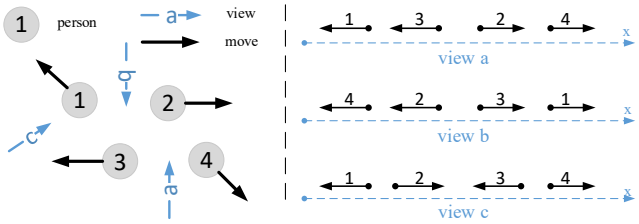


Figure 7. Ordering as view changes. We define moving left and right as $-1, +1$ respectively and want to recognize the activity *dismissal*. Absolute ordering is not important (view a&b). The importance is that *dismissal* must be represented as sequence $\{-1, -1, +1, +1\}$ in x coordinate. For view c, it may be confusing but it will have same pattern as view a&b as person moves.

Usually, views in surveillance videos don't change so much.

There are also some limitations of our method performed on Collective Activity Dataset [6]. Some of them are due to dataset. In the dataset, the amount of data is small and the diversity is poor. There are also some inaccurate annotations (*e.g.*, confused annotations in the transition of two type of action), and it easily leads to misunderstanding. We use sub-action data augmentation to compensate it but can not avoid it completely. In addition, for atomic action classification, the performance is mainly limited by tracking and optical flow algorithm. It is hard to generate optical flow images when people walk near buildings covered by shadows. Moreover, the threshold used in encoder on context need to be set via cross-validation.

5. Conclusions

In this paper, we focus on learning the multi-level interaction context and develop a hierarchical recurrent interactional context encoding framework for collective activity recognition. LSTM based feature aggregation method is employed to model the action of majority and context encoder is used to generate multi-level interaction context. Results show the success of learning context information based on LSTM with weak label and validate the encoding-decoding ability of LSTM. Furthermore, the proposed method is powerful for recognizing group activity, robustness to noisy human detections and flexible enough to model high order interactional context.

6. Acknowledgement

The work was supported by State Key Research and Development Program (2016YFB1001003). This work was partly supported by NSFC (61502301), China's Thousand Youth Talents Plan, National Natural Science Foundation of China (61521062), the 111 Project (B07022) and the Shanghai Key Laboratory of Digital Media Processing and Transmissions.

References

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F. Li, and S. Savarese. Social LSTM: human trajectory prediction in crowded spaces. In *CVPR*, 2016.

[2] M. R. Amer and S. Todorovic. A chains model for localizing participants of group activities in videos. In *ICCV*, 2011.

[3] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. *CoRR*, abs/1412.7755, 2014.

[4] X. Chang, W. S. Zheng, and J. Zhang. Learning person person interaction in collective activity recognition. *TIP*, 24(6), 2015.

[5] W. Choi and S. Savarese. A unified framework for multi-target tracking and collective activity recognition. In *ECCV*, 2012.

[6] W. Choi, K. Shahid, and S. Savarese. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *ICCV*, 2009.

[7] W. Choi, K. Shahid, and S. Savarese. Learning context for collective activity recognition. In *CVPR*, 2011.

[8] Z. Deng, A. Vahdat, H. Hu, and G. Mori. Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition. In *CVPR*, 2016.

[9] Z. Deng, M. Zhai, L. Chen, Y. Liu, S. Muralidharan, M. J. Roshtkhari, and G. Mori. Deep structured models for group activity recognition. In *BMVC*, 2015.

[10] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[11] A. Dosovitskiy, P. Fischery, E. Ilg, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox, et al. Flownet: Learning optical flow with convolutional networks. In *ICCV*, 2015.

[12] G. Gkioxari and J. Malik. Finding action tubes. In *CVPR*, 2015.

[13] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*, 2014.

[14] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*, 2013.

[15] H. Hajimirsadeghi, W. Yan, A. Vahdat, and G. Mori. Visual recognition by counting instances: A multi-instance cardinality potential kernel. In *CVPR*, 2015.

[16] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8), 1997.

[17] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. A hierarchical deep temporal model for group activity recognition. In *CVPR*, 2016.

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

[19] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[21] T. Lan, L. Sigal, and G. Mori. Social roles in hierarchical models for human activity recognition. In *CVPR*, 2012.

[22] T. Lan, Y. Wang, W. Yang, S. N. Robinovitch, and G. Mori. Discriminative latent models for recognizing contextual group activities. *PAMI*, 34(8), 2012.

[23] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu. Recurrent models of visual attention. In *NIPS*, 2014.

[24] B. Ni, S. Yan, and A. Kassim. Recognizing human group activities with localized causalities. In *CVPR*, 2009.

[25] B. Ni, X. Yang, and S. Gao. Progressively parsing interactional objects for fine grained action detection. In *CVPR*, 2016.

[26] V. Ramanathan, B. Yao, and L. Fei-Fei. Social role discovery in human events. In *ICCV*, 2013.

[27] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. In *CVPR*, 2012.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3), 2015.

[29] P. Sermanet, A. Frome, and E. Real. Attention for fine-grained categorization. *CoRR*, abs/1412.7054, 2014.

[30] T. Shu, D. Xie, B. Rothrock, S. Todorovic, and S.-C. Zhu. Joint inference of groups, events and human roles in aerial videos. In *CVPR*, 2015.

[31] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 2014.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[33] M. Sundermeyer, R. Schlüter, and H. Ney. Lstm neural networks for language modeling. In *Interspeech*, 2012.

[34] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.

[35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[36] V. Veeriah, N. Zhuang, and G.-J. Qi. Differential recurrent neural networks for action recognition. In *ICCV*, 2015.

[37] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.

[38] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *ACM MM*, 2015.