An Actor-centric Causality Graph for Asynchronous Temporal Inference in Group Activity

Zhao Xie, Tian Gao, Kewei Wu*, Jiao Chang
Key Laboratory of Knowledge Engineering with Big Data, Hefei University of Technology
School of Computer Science and Information Engineering, Hefei University of Technology
xiezhao@hfut.edu.cn; gaotian@mail.hfut.edu.cn; wukewei@hfut.edu.cn; jiaochang@mail.hfut.edu.cn

Abstract

The causality relation modeling remains a challenging task for group activity recognition. The causality relations describe the influence on the centric actor (effect actor) from its correlative actors (cause actors). Most existing graph models focus on learning the actor relation with synchronous temporal features, which is insufficient to deal with the causality relation with asynchronous temporal features. In this paper, we propose an Actor-Centric Causality Graph Model, which learns the asynchronous temporal causality relation with three modules, i.e., an asynchronous temporal causality relation detection module, a causality feature fusion module, and a causality relation graph inference module. First, given a centric actor and its correlative actor, we analyze their influences to detect causality relation. We estimate the self influence of the centric actor with self regression. We estimate the correlative influence from the correlative actor to the centric actor with correlative regression, which uses asynchronous features at different timestamps. Second, we synchronize the two action features by estimating the temporal delay between the cause action and the effect action. The synchronized features are used to enhance the feature of the effect action with a channel-wise fusion. Third, we describe the nodes (actors) with causality features and learn the edges by fusing the causality relation with the appearance relation and distance relation. The causality relation graph inference provides crucial features of effect action, which are complementary to the base model using synchronous relation inference. Experiments show that our method achieves state-of-the-art performance on the Volleyball dataset and Collective Activity dataset.

1. Introduction

Group activity recognition is a challenging task to identify the group activity by analyzing the actors that perform different actions. It has been widely used in many applications in video surveillance, social interaction analysis, and sports analysis [4, 14, 25, 40]. Unlike individual action recognition, group activity recognition learns the relation between actors to infer the group activity. The relation between two actors can be explained as cause-effect relation (denoted as causality relation), in which the action of one

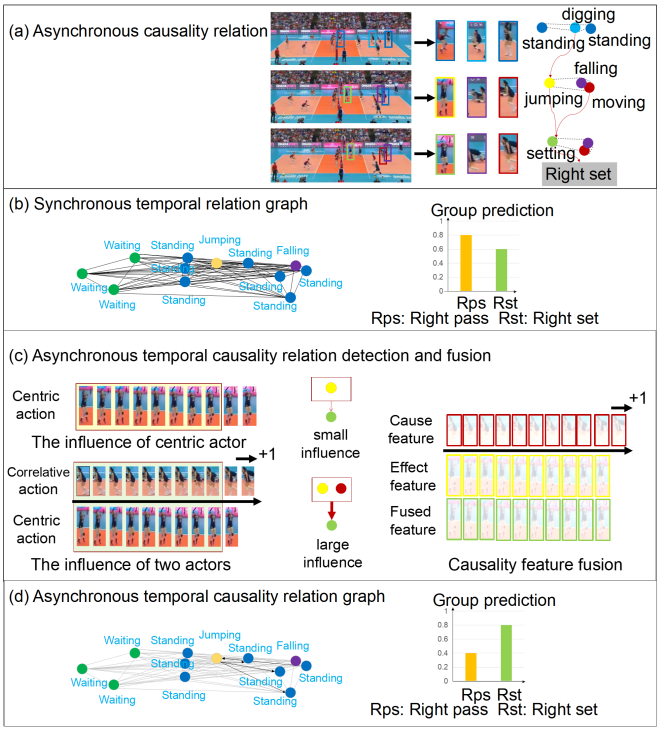

Figure 1. Illustration of the causality relation. (a) Asynchronous causality relation. (b) Synchronous temporal relation graph. (c) Asynchronous temporal causality relation detection and fusion. (d) Asynchronous temporal causality relation graph learns the influences of two actors to detect causality relation. In this work, we enforce the graph model with asynchronous causality relation by analyzing the actors' influences.

*Corresponding author.

actor (denoted as the cause actor) impacts the action of another actor (denoted as the effect actor). The two actors impact each other individually resulting in different causality relations. As the effect action performs after the cause action with a temporal delay, the asynchronous temporal features of the two actors hinder causality relation learning. Existing methods describe the relation with the appearance feature [15] and the position feature [24, 36]. The above methods merely learn the spatial relation in each frame, which neglect to describe the relation with the temporal dynamics in the frame sequence. Some methods describe the relation with the temporal feature learned by RNN [28] and Transformer network [18]. The existing methods always learn the relation at the same timestamp with the synchronous temporal feature, and neglect to describe the influences of two actors, who have asynchronous temporal relation. Therefore, it is still challenging to capture the relation with asynchronous temporal features for group activity recognition.

As the causality relation is asynchronous, we decompose our graph model with asynchronous causality relation into two sub-tasks illustrated in Figure 1b: (1) the causality relation detection task by analyzing the influences of two actors with their asynchronous features, and (2) the asynchronous causality feature fusion task by integrating the feature of the effect actor with the synchronized feature of the cause actor.

Figure 1 shows the group activity prediction influenced by different actors. (1) In Figure 1a, with a temporal delay after the cause action performs (digging), the cause actor changes to the action "falling", and the effect actor changes to the action "jumping". Actors change their states with a temporal delay in an asynchronous way, which hinders cause-effect (causality) relation learning. (2) As shown in Figure 1b, the traditional method addresses the group activity using a synchronous temporal relation graph, which contains a large number of irrelevant relations. The synchronous graph is hard to detect the cause-effect relation with the synchronous features at a single timestamp. Without learning the asynchronous causality relation, the model can not explain why the effect actor jumps. For example, it uses the falling cause actor to mispredict group activity as the "Right pass". (3) To detect the causality relations between the centric actor and its correlative actor, we focus on learning the influences with their asynchronous features. When the centric actor is affected by the correlative actor, the influence of two actors is larger than the influence of the centric actor itself. The influences of two actors can detect the causality relation from the correlative actor to the centric actor. As shown in Figure 1c, after the correlative actor moves, the centric actor jumps with the temporal delay of one frame (+1). Then, the causality relation is used to enhance the centric actor features by fusing two actors' features. (4) When we learn asynchronous causality relations to form a causality relation graph, which can select the relevant edges and enhance node features. In Figure 1d, the causality inference process finds the relation from the moving actor to the jumping actor, and helps to explain the actor jumps for setting the volleyball. The causality relation analyzes the influences learned with the asynchronous temporal features, which is complementary to the relation learned with synchronous temporal features. The framework by integrating two relation graphs can successfully predict the group activity as the "Right set".

In this paper, we propose an Actor-Centric Causality Graph (ACCG) Model to detect the asynchronous causality relation for group activity recognition. Figure 2 shows the overview of the proposed model. The model consists of three modules, i.e., an asynchronous temporal causality relation detection module, a causality feature fusion module, and a causality relation inference module. First, we detect the causality relation between the centric actor and its correlative actor by analyzing the self influence and the correlative influence. We learn the self influence with self regression, and learn the correlative influence with correlative regression. We extend the correlative regression with asynchronous features of the correlative actor, which helps to learn the asynchronous causality relation by analyzing the influences of the two actors. Second, the temporal delay of the causality relation is estimated to synchronize two action features. We integrate them with a channel-wise fusion to learn the causality feature of the effect actor. Third, we describe the actors (nodes) with asynchronous causality features, and describe the edges with the causality relation for graph inference. The causality relation inference provides the crucial features of actors, which are complementary to synchronous relation inference. We apply the base model to learn the synchronous relation inference, and add two relation inferences to enhance the group relation learning.

Our contributions are summarized as follows:

(1) We propose an Actor-Centric Causality Graph Model, which detects the asynchronous causality relation by analyzing the influences of two actors at different timestamps. We design the self regression to estimate the self influence of the centric actor. We design the corelative regression with the asynchronous features of two actors to estimate their correlative influence.

(2) We design a causality feature fusion to enhance the feature of the centric actor by integrating it with the synchronized feature of its correlative actor. The synchronized feature is generated by estimating the temporal delay between the asynchronous features of two actors.

(3) Our Actor-Centric Causality Graph Model learns the asynchronous relation, which is complementary to synchronous relation learning. Our framework integrates two relations and achieves state-of-the-art performance on the Volleyball dataset and Collective Activity dataset.

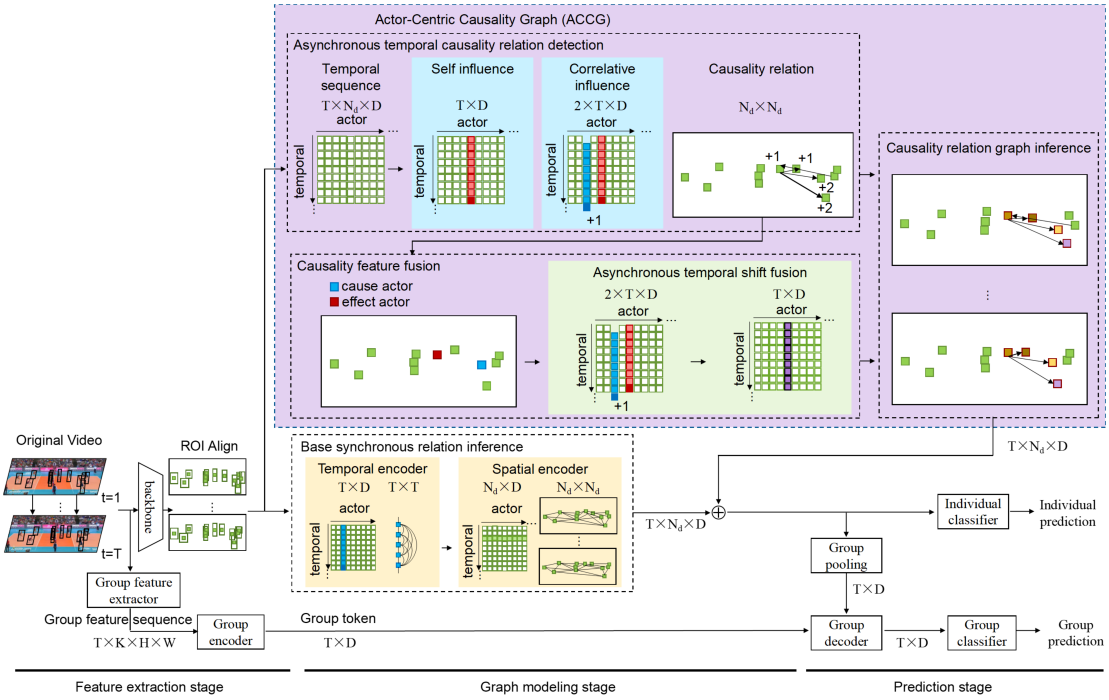

Figure 2. The overview of the actor-centric causality graph model. The base model learns the relation with synchronous features at each timestamp. The actor-centric causality graph is proposed to analyze the influence of two actors with asynchronous features, which can learn the causality relation in the asynchronous temporal causality relation detection module. The causality feature fusion model enhances the centric action with synchronized correlative action features. The causality relation graph inference module learns the contextual feature with causality relation. The framework combines the causality relation graph and the base graph for better graph relation learning.

2. Related Work

Group Activity Recognition. Group activity recognition learns the group feature by organizing the actor feature with relation. Existing methods use graph models to learn the relation between actors with appearance features [1, 15] and position features [24, 36]. Some methods learn the relation adaptively by introducing self-attention [10, 23] and multi-head attention [8, 18]. Some methods learn the relation by selecting the key actors [22]. The above methods learn the relation with spatial features. Some methods learn the relation with temporal features [11–13, 17, 26–28, 33]. These methods use the feature at the same timestamp, and neglect to learn the relation by analyzing the asynchronous temporal features.

Granger Causality Test. The causality test describes the cause-effect relation of two things [39]. Some methods learn the cause-effect relation with and-or graph [7], and bayesian inference [6]. The Granger causality test learns the cause-effect relation by analyzing the feature residual estimated with temporal feature regression [9] and extends to lagged temporal analysis [5, 29], reversal temporal analysis [35], and selected feature analysis [3]. The Granger causality test has not been used to analyze the relation between two actors for group activity recognition.

Asynchronous Temporal Modeling. The asynchronous features have been studied for the temporal feature integration [16, 34]. The asynchronous features can be fused with multiple temporal features [30] and multiple modality features, including appearance feature and motion feature [20]. The asynchronous features have been exploited with spatial-temporal asynchronous normalization [21]. Unlike the above methods, we use asynchronous features to model the feature reconstruction of two actors in the group activity. The feature reconstruction can learn the influences of them, and can be analyzed with the Granger causality test to detect causality relation.

3. Base Group Activity Recognition Model

Most existing methods predict the group activity with three main stages as shown in Figure 2 [18, 25, 36, 38]. (1) The feature extraction stage extracts features of the frame sequence using the pre-trained backbone model. (2) The graph modeling stage learns the relation to provide contextual features of the actors. (3) The prediction stage generates the actor action label and the group activity label and computes the prediction loss for training. In the following, we review the common practices of the graph relation model, which is used as our base model.

3.1. Feature Extraction

Following [18, 36], we use Inception-v3 to extract a feature map for a video sequence, and use RoIAlign to extract the features of the actor's bounding boxes. After that, an FC layer is performed on the aligned features to get the feature map for actors $X \in \mathbb{R}^{T \times N_d \times D}$. T, N_d, and D denote frame number, actor number, and feature dimension.

Following [18], the group feature extractor uses the backbone model to generate the feature map of each frame. 2D convolution is applied to learn the group feature sequence $X_g \in \mathbb{R}^{T \times K \times H \times W}$. K, H, and W denote the token number, the height, and the width of the group feature. The group encoder reshapes the feature into a flattened feature and uses a softmax operation to generate the spatial attention matrix. 2D convolution is used to project the group feature into D channels, which is used to learn K tokens by computing the weighted summation of every pixel with spatial attention. Another average pooling is used to get the group token $X_G \in \mathbb{R}^{T \times D}$.

3.2. Spatial-Temporal Graph

Following [18], the base synchronous relation can be learned with a temporal encoder and a spatial encoder in a stacked manner. The temporal encoder adopts a transformer-based model to learn the temporal relation of each actor. The input actor feature sequence is split to get the feature sequence of each actor $X_i \in \mathbb{R}^{T \times D}$, where i is the actor index. The transformer-based temporal feature is estimated as:

$$\begin{cases} Q_i = X_i W_Q, K_i = X_i W_K, V_i = X_i W_V, \\ V_i' = softmax(\frac{Q_i K_i^T}{\sqrt{D}}) V_i + V_i \\ V_i'' = FFN(V_i') \end{cases} \quad (1)$$

where W_Q, W_K, W_V are learnable parameters shaped as $D \times D$. FFN is the feed-forward network in the canonical Transformer. The features of all actors are packed together to the temporal features $V_T \in \mathbb{R}^{N_d \times T \times D}$.

The spatial encoder learns the spatial relation between actors. The spatial encoder follows the operation of the temporal encoder. The difference with the above temporal encoder is that the spatial encoder uses the features at each timestamp, which is split from the output of the temporal encoder. We pack the output of the spatial encoder to get the features of all frames $V_S \in \mathbb{R}^{T \times N_d \times D}$.

3.3. Group Activity Prediction and Training Losses

The base model predicts the actor action scores and group activity scores. For actor action recognition, a classifier with two FC layers takes the learned actor features as input to predict each actor action score. For group activity recognition, a group decoder [18] is applied to predict the group activity. The group decoder takes the group token as the group query. The decoder takes the group pooling features from individual features as the group key. The updated group query is learned by summarizing the group query with the overall context from the group pooling features. The group classifier uses two FC layers to predict the group activity scores.

Given the ground truth labels for actor action and group activity, the loss function considers the actor action recognition loss, the group activity recognition loss, and the relation contrastive loss following [25]. The contrastive loss compares the feature similarity between the nodes in the same relation graph and the similarity between the nodes in different relation graphs. The contrastive loss encourages the diversity of relation learning.

3.4. Discussion

The base model uses a transformer-based temporal encoder, which can enhance the temporal features of actors. The base model learns the spatial relation with the synchronous temporal features at the same timestamp, and neglects to analyze the influence of two actors with asynchronous features. The main focus of our work is to analyze the influence of two actors with asynchronous features, which helps to detect the asynchronous causality relation for relation learning, as described in the following section.

4. Actor-Centric Causality Graph Model

Figure 2 shows the overview of our proposed methods by embedding the actor-centric causality graph in the graph modeling stage. The graph learns the asynchronous causality relation, which is complementary to the relation with synchronous temporal features. We represent the actor-centric causality graph as $G^{cau} = \{V^{cau}, E^{cau}\}$. The causality relations are learned by analyzing the influences of two actors in the Asynchronous Temporal Causality Detection Module. The nodes V^{cau} describe the causality fused features of actors, which are learned with the Causality Features Fusion Module. The edges E^{cau} detail the causality relations with appearance relations and distance relations, which are used for graph relation reasoning in the causality relation inference module.

4.1. Asynchronous Temporal Causality Relation Detection Module

The causality relation indicates the cause actor and effect actor. We introduce feature reconstruction to learn the reconstruction residual, which can be used to estimate the action influence. The effect actor takes a large influence from the cause actor, and its action has a large residual of feature reconstruction with its historical features. When the feature reconstruction considers the historical features of the cause

action and effect action, the residual of effect action reconstruction becomes small. To detect the causality relation, we select the centric actor and correlative actor. To learn the self influence, we design self regression to model the feature reconstruction with the historical features of the centric actor. To learn the correlative influence, we design corelative regression to model the feature reconstruction with the historical features of the two actors. The large difference between the two influences detects a strong causality relation between the two actors.

Self Influence Estimation. We estimate the self influence with the residual between the frame feature and the reconstructed feature, which is reconstructed with the historical feature of the actor itself. Given the feature sequence $X_i \in \mathbb{R}^{T\times D}$, We design self regression to reconstruct the current frame $\hat{x}_k^i$ using the historical feature bank with the time window $[k-m, k-1]$. m is the window size. The self influence ssr^i is estimated with the sum of squares residual (SSR) as:

$$\begin{cases} \hat{x}_k^i = \sum\limits_{r=k-m}^{k-1} \omega_r^i x_r^i + b^i \\ ssr^i = \sum\limits_k \left\| x_k^i - \hat{x}_k^i \right\|_2^2 \end{cases} \tag{2}$$

where ω_r^i, b^i are the parameters in self regression. The parameters are learned with the sum of the squared reconstruction error $\mathcal{L}_{self} = ssr^i$.

Correlative Influence Estimation. We estimate the correlative influence using the construction model with two actors' features. We design the correlative regression for feature construction. We extend it to asynchronous correlative regression by temporally shifting the features of the correlative features with the temporal delay. The temporal delay explains the effect action occurs after the cause action. Given the correlative actor features X_j and the temporal delay $delay$, the asynchronous historical time window of the correlative actor is $[k - delay - m, k - delay - 1]$. The asynchronous historical features of two actors are used to estimate the reconstruction feature of the centric actor $\hat{x}_k^{j\to i}$. The correlative influence $srr^{j\to i}$ is estimated with the sum of squares residual (SSR) as:

$$\begin{cases} \hat{x}_k^{j\to i} = \sum\limits_{r=k-m}^{k-1} \omega_r^{j\to i} x_r^i + \sum\limits_{r'=k-delay-m}^{k-delay-1} \omega_{r'}^{j\to i} x_{r'}^j + b^{j\to i} \\ ssr^{j\to i} = \sum\limits_k \|x_k^i - \hat{x}_k^{j\to i}\|_2^2 \end{cases} \tag{3}$$

where $\omega_r^{j\to i}$, $\omega_{r'}^{j\to i}$, $b^{j\to i}$ are the parameters in corelative regression. The parameters are learned with the sum of the squared reconstruction error $\mathcal{L}_{corr} = ssr^{j\to i}$.

Granger Causality Relation Estimation. We estimate the causality relation by analyzing the two influences with the Granger causality test. The Granger causality test explains the distribution of the causality relation over the difference between two influences. The actor features are supposed to obey the Gaussian distribution, which can imply the influence, i.e. sum of squares residual (SSR) of the reconstructed feature, obeys the χ^2 distribution. Granger causality test uses the test statistic $f^{j\to i}$ to analyze two χ^2 distributions as:

$$f_{j\to i} = \frac{\left(ssr^{j\to i} - ssr^i\right)/m}{ssr^i/\left(n_m - v_m\right)} \tag{4}$$

The test statistic obeys the Fisher-Snedecor distribution, which has two degrees of freedom. $n_m = (T-m)D$ is the sample number. $v_m = 2m+1$ is the degrees of freedom of the corelative regression. The Fisher-Snedecor distribution $\psi_F(\cdot)$ can project the test statistic into the causality probability value as:

$$p_{j\to i} = \int_0^{f_{j\to i}} \psi_{F(m, n_m - v_m)}(z)dz \tag{5}$$

Our graph focuses on learning multiple causality relations by considering multiple asynchronous temporal features. Multiple asynchronous temporal features are learned by temporal shifting with different temporal delays in the correlative regression. The causality relation with the largest probability indicates the estimated temporal delay $delay^*_{j\to i}$ and the estimated causality probability $p_{j\to i}$ between two actions as:

$$[delay^*_{j\to i}, p^*_{j\to i}] = argmax_{delay}\, p_{j\to i} \tag{6}$$

Our graph represents the causality relations $A^{Granger} = \{a^{\text{Granger}}_{j\to i}\}$. Each causality relation between two actors is detected by comparing the causality probability with a causality threshold as:

$$a^{\text{Granger}}_{j\to i} = \begin{cases} 1, & p^*_{j\to i} > \tau \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

4.2. Causality Features Fusion Module

The causality relation indicates the cause actor and the effect actor in the group activity. The effect action can be enhanced by integrating it with the cause action. The features of the cause actor are synchronized by temporally shifting to the effect actor with the estimated delay. Due to we cannot observe the features before the input frames, the early part of the shifted cause action features cannot be directly estimated. We fill this early part using the feature of the first frame. The temporal shift operation is formed as:

$$x_k^{shift, j\to i} = \begin{cases} x_1^j, & k - delay^*_{j\to i} < 1 \\ x^j_{k-delay^*_{j\to i}}, & otherwise \end{cases} \tag{8}$$

We use channel-wise concatenation to integrate the effect action features with the shifted cause action features. The impact of each part is considered by introducing the channel ratio parameter d. The input cause action feature has the shape $\mathbb{R}^D$. The cause action feature is projected with the parameter w_i^d in an FC layer to the shape $\mathbb{R}^{D/d}$. The effect action feature is projected with the parameter $w^d_{j\to i}$ to

the shape $\mathbb{R}^{D-D/d}$. We concatenate them and analyze the impact of the cause action feature by adjusting the parameter in two projection layers as follows:

$$x_k^{con,j\to i} = concat(w_i^d x_k^i, w_{j\to i}^d x_k^{shift,j\to i}) \tag{9}$$

In the group activity, the centric actor may have multiple causality relations to different cause actors. Each causality relation indicates a cause action to learn effect action features separately. We average the features learned with multiple causality relations as:

$$x_k^{syn,i} = \frac{1}{\sum_j a_{j\to i}^{Granger}} \sum_j x_k^{con,j\to i} \tag{10}$$

Our graph uses the node representation of all actors as $V^{cau} = \{X_i^{syn}\}$, where each actor has the feature $X_i^{syn} = \{x_k^{syn,i}\}$.

4.3. Causality Relation Graph Inference Module

Our graph represents nodes with causality fusion features and represents the edges with causality relations. The nodes describe effect action with asynchronous fused features by detecting the temporal delay, which is complementary to the synchronous features at the same timestamp learned with the temporal transformer. The edges using causality relations can enhance the relevant relations in the spatial transformer. To detail the causality relation with the actor's appearance and position, we embed the causality relation with the appearance relation and distance relation.

Following [36], h graphs are used to learn appearance relations, which is estimated by the dot product of two actor's features as: $a_{i,j,h}^{app} = \frac{(w_h^i x^i)^T (w_h^j x^j)}{\sqrt{D}}$. The distance relation considers s masks with different distance ratios λ_s to limit the relation based on the image width $width$. When the distance between actor i and actor j is smaller than the distance ratio $dist_{i,j} \leq \lambda_s width$, the distance relation is $a_{i,j,s}^{dist} = 1$. Otherwise, the distance relation is 0. Our graph embeds the causality relation with the appearance relation and distance relation to learn the causality edges $E_{h,s}^{cau} = \{e_{h,s}^{j\to i}\}$ as:

$$e_{h,s}^{j\to i} = \frac{a_{j\to i}^{Granger} a_{i,j,h}^{app} a_{i,j,s}^{dist}}{\sum_j a_{j\to i}^{Granger} a_{i,j,h}^{app} a_{i,j,s}^{dist}} \tag{11}$$

Our graph learns the output features of nodes with multiple edges as:

$$X' = \sum_{h,s} ReLU(E_{h,s}^{cau} V^{cau} W_{h,s}^{graph}) \tag{12}$$

The output features of our actor-centric causality graph consider the asynchronous temporal causality relation. The graph in the base model considers the synchronous temporal relation with the transformer-based network. Two graphs are added together to enhance the actor representation.

5. Experiments

5.1. Datasets and Implementation Details

Datasets. We conduct experiments on two widely-adopted group activity datasets which contain tracking annotations and bounding boxes, including the Volleyball dataset [14]and the Collective Activity dataset [4]. The metric employs the Multi-class Classification Accuracy [36].

The Volleyball dataset [14] contains 55 video recordings of volleyball games and is split into 3493 training clips and 1337 testing clips. The center frame of each clip is annotated with bounding box coordinates for all actors and their action labels (i.e. blocking, digging, falling, jumping, moving, setting, spiking, standing, and waiting). Each clip is annotated with one group activity label out of eight labels (i.e. right set, right spike, right pass, right winpoint, left set, left spike, left pass, and left winpoint).

The Collective Activity dataset [4] contains 44 clips. We follow the train set and test set in [36]. The center frame of every ten frames is annotated with bounding box coordinates of all actors and their action labels (i.e. NA, crossing, waiting, queueing, walking, and talking). Every ten frames are given one group activity label out of five (i.e. crossing, waiting, queueing, walking, and talking).

Implemenation Details. For feature extraction, we adopt the ImageNet pre-trained Inception-v3 [32] as the backbone. The RoIAlign [36] is applied to extract the actor feature with each bounding box. The feature is embedded into $D = 1024$ channels with an FC layer. Besides the Inception-v3 features, we consider the pose feature with the ImageNet pre-trained HRNet-48 as backbone [31]. The pose feature is extracted with each bounding box and concatenates with the Inception-v3 feature to enhance the actor feature. The concatenated feature is embedded into D channels with an FC layer. Following [18], the base model takes the group token number $K = 8$ in the group encoder. The spatial encoder and temporal encoder use 8 attention heads. In the Actor Centric Causality Graph, we set the window size $m = 4$, the set of temporal delay $delay = \{0, 1, 2\}$, the causality threshold $\tau = 0.9$, the channel ratio parameter $d = 6$. For causality relation graph inference, we set the appearance graph number $k = 16$, and the set of distance ratios $\lambda_s = \{0.1, 0.2, 0.3, 0.4\}$.

We train our model in three stages. First, the parameters in the base model are trained with the base model loss. Second, the parameters in the Asynchronous Granger Causality Detection Module, including the self-regression and correlative regression, are trained in each clip. Third, the framework combines our causality relation graph and the base graph.

In the first and third training stage, the stochastic gradient descent with ADAM is adopted as an optimizer with fixed hyper-parameters to β_1 = 0.9, β_2 = 0.999, $epsilon$

Table 1. Comparisons with the state-of-the-art methods on Volleyball dataset and Collective Activity dataset.

Method	Flow	Backbone	Volleyball		Collective Activity
			Group	Individual	Group
HDTM [14]		AlexNet	81.9	-	81.5
CERN [27]		VGG16	83.3	-	87.2
CCGLSTM [33]		AlexNet	89.3	-	93.0
HRN [15]		VGG19	89.5	-	-
SSU [2]		Inception-v3	90.6	81.8	-
PRL [13]		VGG16	91.4	-	93.8
PDAR [24]		Inception-v3	92.2	-	90.3
ARG [36]		Inception-v3	92.5	82.8	91.0
AT [8]	✓	I3D	93.0	83.7	92.8
GLIL [28]		Inception-v3	93.0	-	94.9
VC [37]		Inception-v3	93.3	-	95.1
CRM [1]	✓	I3D	93.0	-	85.8
DIN [38]		VGG16	93.6	-	95.9
GF [18]		Inception-v3	94.1	83.7	93.6
GRAIN [19]		VGG16	94.5	-	95.2
SAACRF [25]	✓	I3D+HRNet	96.4	85.5	96.0
Base model		Inception-v3	93.6	83.8	92.6
Base+ACCG		Inception-v3	95.0	85.6	94.5
Base+ACCG		VGG-16	95.5	85.8	95.0
Base+ACCG	✓	I3D+HRNet	**96.7**	**86.4**	**96.3**

Table 2. The effect of the temporal delay setting.

Delay	Group		Individual	
	wo shift	adaptive	wo shift	adaptive
{0}	93.6	-	84.5	-
{0,1}	94.4	94.7	85.1	85.3
{0,1,2}	**94.5**	**95.0**	**85.2**	**85.6**
{0,1,2,3}	**94.5**	**95.0**	**85.2**	**85.6**

$=10^{-8}$. For the Volleyball dataset, we train the network in 150 epochs in each stage with a mini-batch size of 32 and a dropout ratio of 0.3 [36]. For the Collective Activity dataset, we train the network in 80 epochs in each stage with a mini-batch size of 16 and a dropout ratio of 0.5 [36]. For both datasets, the initial learning rate is 0.00001 and decreases by 0.1 after every 40 iterations.

In model prediction, we use each test video clip to update the parameters of the self-regression and correlative regression, and fix the parameters of the rest modules.

5.2. Comparison with the State-of-the-Art

Volleyball Dataset. Table 1 shows the performance results on the Volleyball dataset. The base model is reported without considering the influences of actors. Our model embeds the ACCG to analyze the influences of two actors with asynchronous features, which can detect asynchronous causality relations. The asynchronous causality relation is complementary information to the synchronous temporal relation. The model adds ACCG features and Base features to boost the performance. The Base+ACCG outperforms the model with spatial-temporal relation learned in GF [18]. Using the Flow, I3D, and HRNet [31] features, the Base+ACCG outperforms the SAACRF [25].

Collective Activity Dataset. Table 1 shows the performance results on the Collective Activity dataset. The Base+ACCG model considers both the synchronous relation and the asynchronous causality relation learned by analyzing the influences of actors. The Base+ACCG model with RGB feature outperforms the GF method [18].

Table 3. The effect of channel ratio parameter.

Channel ratio parameter	Group	Individual
without	93.3	83.7
d=2	94.1	84.7
d=4	94.6	85.3
d=6	**95.0**	**85.6**
d=8	93.5	84.0

Table 4. The effect of graph number and distance mask number.

Appearance graph	Group			Individual		
	Distance mask			Distance mask		
	2	3	4	2	3	4
8	93.5	94.5	94.8	84.1	85.1	85.6
16	**93.7**	**94.7**	**95.0**	**84.1**	**85.1**	**85.6**
32	93.6	94.6	94.9	84.0	85.0	85.5

Table 5. The effect of graph attention.

Method	Volleyball	
	Group	Individual
ARG [36]	92.5	82.8
ACCG wo attention	93.3	83.1
ACCG	93.6	83.7
ARG+ACCG	**93.9**	**84.0**
Base model	93.6	83.8
Base+ACCG wo attention	94.3	84.7
Base+ACCG	**95.0**	**85.6**

Table 6. Comparison with other graph relation learning. GD, SE, and TE denote the Group decoder, Spatial encoder, and Temporal encoder in the base model. V^{cau}, E^{cau} denote the node representation and edge relation in the causality relation graph.

Base model				Causality graph		Group	Individual
Inception-v3	GD	SE	TE	V^{cau}	E^{cau}		
✓						89.8	80.9
✓	✓					91.0	82.1
✓	✓	✓				91.8	82.2
✓	✓	✓	✓			93.6	83.8
✓				✓		91.9	82.5
✓					✓	92.9	83.3
✓				✓	✓	93.6	83.7
✓	✓	✓	✓	✓	✓	**95.0**	**85.6**

5.3. Ablation Study

To dispel any concerns that the improvement is simply from additional optical flow and pose information, we perform ablation studies using RGB features learned with the Inception-v3 backbone on the Volleyball dataset.

The effect of the temporal delay setting. Table 2 analyzes the model with different temporal delays. We consider multiple asynchronous temporal features by setting multiple temporal delays. In Table 2, the model without temporal shifting (wo shift) learns the causality feature with the original correlative features. In Table 2, the model with adaptive temporal shifting (adaptive) learns the causality feature with the synchronized correlative features. The model with

adaptive temporal shifting outperforms the model without temporal shifting.

The effect of channel ratio parameter. Table 3 analyzes the model with different channel ratio parameters, which are used to adjust the impact of the cause action and the effect action. The model without considering the cause action features gets the worst performance. We introduce the channel ratio parameter d to enhance the effect action features with cause action features. We increase the channel ratio parameter to enlarge the impact of effect action and get the best performance at $d = 6$.

The effect of appearance graph number and distance mask number. Table 4 analyzes the model with multiple appearance graphs in the graph relation inference. The model with 16 appearance graphs gets enough appearance relations. For distance relation learning, we adopt 4 distance masks. The model with 4 distance mask and 16 appearance graphs get the best performance.

The effect of graph attention. Table 5 provides the ACCG with/without graph attention. ACCG model considers ARG [32] relation, which uses graph attention to learn the appearance relation. ACCG uses asynchronous temporal relations to select crucial ARG relations. We provide ACCG without (wo) attention by removing the ARG relation from ACCG. ACCG wo attention uses a set of temporal delays to capture asynchronous temporal relations, and outperforms the ARG model. We provide a two-branch model ARG+ACCG to adopt the ARG as the base model. ARG+ACCG integrates two branch relations in a complementary way, and outperforms ARG and ACCG.

Comparision with other graph relation learning. Table 6 analyzes the contribution of each component. The base model uses GD, SE, and TE to learn the actor relation and improve performance. In the causality graph, our model enhances the node features by fusing two actors' features by synchronizing with the temporal delay detected in the causality relation. The causality fused features help to increase the performance. Our model learns asynchronous causality relations to indicate relevant relations in the group activity for better relation learning. Without considering the synchronous relation learned in the base model, our causality graph with asynchronous causality relation outperform the base model. The asynchronous causality relation is complementary to the transformer-based relation. We integrate the two relations to get the best performance. The Base model has 63.6 MParams and 408.5 GFLOPs. The Base+ACCG has 89.8 MParams and 414.8 GFLOPs.

5.4. Visualization

Figure 3 shows two examples of the Volleyball dataset. In (a), The base model considers the moving actor (No.6) as passing the volleyball and mispredicts the group activity as a "Right pass". Our actor-centric causality graph estimates the correlative influence of two actors (No.7 and No.8), which is larger than the self influence (No.8). Our graph analyzes these two influences to indicate the causality relation from No.7 to No.8. Our graph also detects the causality relation from No.9 to No.8. The two causality relations enhance the action of No.8 to correct its prediction as the setting. Besides, our causality relation explains that No.2 is waiting for the volleyball which passed from No.8. In (b), our graph detects two causality relations, which explain that No.11 is Digging the ball passed from No.3 and No.11 is adjusting the action to pass the ball to No.7. Besides, our causality relation explains that No.10 is moving to spike the volleyball which is set by No.11.

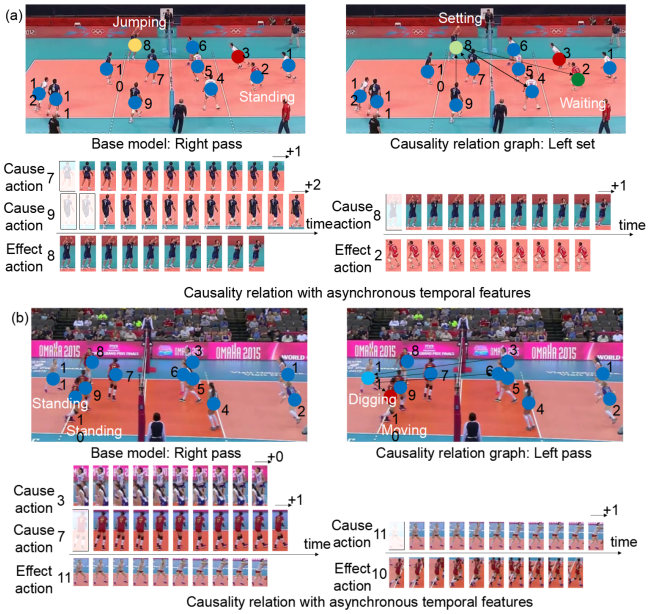

Figure 3. Visualization of the causality relation in the group activity. The changed actor labels are visualized.

6. Conclusion

In this work, we propose an actor-centric causality graph, which focuses on analyzing the influence of two actors for asynchronous causality relation detection. We learn the causality fused feature by integrating the effect action features with the synchronized cause action features. The learned asynchronous relation is complementary to the synchronous relation learned in the transformer-based model. We integrate two relation models, which can outperform the state-of-the-art methods.

7. Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grant 62272144.

References

[1] Sina Mokhtarzadeh Azar, Mina Ghadimi Atigh, Ahmad Nickabadi, and Alexandre Alahi. Convolutional relational machine for group activity recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 7892–7901, 2019. 3, 7

[2] Timur Bagautdinov, Alexandre Alahi, François Fleuret, Pascal Fua, and Silvio Savarese. Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4315–4324, 2017. 7

[3] Gautam Bhattacharya, Koushik Ghosh, and Ananda S. Chowdhury. Granger causality driven AHP for feature weighted knn. *Pattern Recognit.*, 66:425–436, 2017. 3

[4] Wongun Choi, Khuram Shahid, and Silvio Savarese. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *12th IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2009, Kyoto, Japan, September 27 - October 4, 2009*, pages 1282–1289. IEEE, 2009. 2, 6

[5] Ribhu Chopra, Chandra Ramabhadra Murthy, and Govindan Rangarajan. Statistical tests for detecting granger causality. *IEEE Trans. Signal Process.*, 66(22):5803–5816, 2018. 3

[6] Mark Edmonds, Xiaojian Ma, Siyuan Qi, Yixin Zhu, Hongjing Lu, and Song-Chun Zhu. Theory-based causal transfer: Integrating instance-level induction and abstract-level structure learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 1283–1291. AAAI Press, 2020. 3

[7] Amy Sue Fire and Song-Chun Zhu. Learning perceptual causality from video. *ACM Trans. Intell. Syst. Technol.*, 7(2):23:1–23:22, 2016. 3

[8] Kirill Gavrilyuk, Ryan Sanford, Mehrsan Javan, and Cees GM Snoek. Actor-transformers for group activity recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 839–848, 2020. 3, 7

[9] Clive Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–38, 1969. 3

[10] Dan Guo, Kun Li, Zheng-Jun Zha, and Meng Wang. Dadnet: Dilated-attention-deformable convnet for crowd counting. In *Proceedings of the 27th ACM international conference on multimedia*, pages 1823–1832, 2019. 3

[11] Dan Guo, Hui Wang, and Meng Wang. Context-aware graph inference with knowledge distillation for visual dialog. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6056–6073, 2021. 3

[12] Dan Guo, Hui Wang, Hanwang Zhang, Zheng-Jun Zha, and Meng Wang. Iterative context-aware graph inference for visual dialog. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10055–10064, 2020. 3

[13] G. Hu, B. Cui, Y. He, and S. Yu. Progressive relation learning for group activity recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 2020. 3, 7

[14] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. A hierarchical deep temporal model for group activity recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 6, 7

[15] M. S. Ibrahim and G. Mori. *Hierarchical Relational Networks for Group Activity Recognition and Retrieval: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part III*. Computer Vision – ECCV 2018, 2018. 2, 3, 7

[16] Jianing Li, Jia Li, Lin Zhu, Xijie Xiang, Tiejun Huang, and Yonghong Tian. Asynchronous spatio-temporal memory network for continuous event-based object detection. *IEEE Trans. Image Process.*, 31:2975–2987, 2022. 3

[17] Kun Li, Dan Guo, and Meng Wang. Proposal-free video grounding with contextual pyramid network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1902–1910, 2021. 3

[18] Shuaicheng Li, Qianggang Cao, Lingbo Liu, Kunlin Yang, Shinan Liu, Jun Hou, and Shuai Yi. Groupformer: Group activity recognition with clustered spatial-temporal transformer. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 13668–13677, 2021. 2, 3, 4, 6, 7

[19] Wei Li, Tianzhao Yang, Xiao Wu, and Zhaoquan Yuan. Learning graph-based residual aggregation network for group activity recognition. In Luc De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 1102–1108. ijcai.org, 2022. 7

[20] Weiyao Lin, Chongyang Zhang, Ke Lu, Bin Sheng, Jianxin Wu, Bingbing Ni, Xin Liu, and Hongkai Xiong. Action recognition with coarse-to-fine deep feature integration and asynchronous fusion. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 7130–7137. AAAI Press, 2018. 3

[21] Mengyuan Liu, Youneng Bao, Yongsheng Liang, and Fanyang Meng. Spatial-temporal asynchronous normalization for unsupervised 3d action representation learning. *IEEE Signal Process. Lett.*, 29:632–636, 2022. 3

[22] Lihua Lu, Yao Lu, and Shunzhou Wang. Learning multi-level interaction relations and feature representations for group activity recognition. In *International Conference on Multimedia Modeling*, pages 617–628. Springer, 2021. 3

[23] L. Lu, R. Yu, H. Di, L. Zhang, and Y. Lu. Gaim: Graph attention based interaction model for collective activity recognition. *IEEE Transactions on Multimedia*, PP(99):1–1, 2019. 3

[24] D. Pei, A. Li, and Y. Wang. *Group Activity Recognition by Exploiting Position Distribution and Appearance Relation*. MultiMedia Modeling, 27th International Conference,

MMM 2021, Prague, Czech Republic, June 22–24, 2021, Proceedings, Part I, 2021. 2, 3, 7

[25] Rizard Renanda Adhi Pramono, Wen-Hsien Fang, and Yie-Tarng Chen. Relational reasoning for group activity recognition via self-attention augmented conditional random field. *IEEE Trans. Image Process.*, 30:8184–8199, 2021. 2, 3, 4, 7

[26] Mengshi Qi, Yunhong Wang, Jie Qin, Annan Li, Jiebo Luo, and Luc Van Gool. stagnet: an attentive semantic rnn for group activity and individual action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(2):549–565, 2019. 3

[27] Tianmin Shu, Sinisa Todorovic, and Song-Chun Zhu. Cern: confidence-energy recurrent network for group activity recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5523–5531, 2017. 3, 7

[28] Xiangbo Shu, Liyan Zhang, Yunlian Sun, and Jinhui Tang. Host-parasite: Graph lstm-in-lstm for group activity recognition. *IEEE transactions on neural networks and learning systems*, 2020. 2, 3, 7

[29] Elsa Siggiridou and Dimitris Kugiumtzis. Granger causality in multivariate time series using a time-ordered restricted vector autoregressive model. *IEEE Trans. Signal Process.*, 64(7):1759–1773, 2016. 3

[30] Gunnar A. Sigurdsson, Santosh Kumar Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5650–5659. IEEE Computer Society, 2017. 3

[31] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5693–5703. Computer Vision Foundation / IEEE, 2019. 6, 7

[32] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society, 2016. 6

[33] Jinhui Tang, Xiangbo Shu, Rui Yan, and Liyan Zhang. Coherence constrained graph LSTM for group activity recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(2):636–647, 2022. 3, 7

[34] Jiajun Tang, Jin Xia, Xinzhi Mu, Bo Pang, and Cewu Lu. Asynchronous interaction aggregation for action detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XV*, volume 12360 of *Lecture Notes in Computer Science*, pages 71–87. Springer, 2020. 3

[35] Irene Winkler, Danny Panknin, Daniel Bartz, Klaus-Robert Müller, and Stefan Haufe. Validity of time reversal for testing granger causality. *IEEE Trans. Signal Process.*, 64(11):2746–2760, 2016. 3

[36] Jianchao Wu, Limin Wang, Li Wang, Jie Guo, and Gangshan Wu. Learning actor relation graphs for group activity recognition. In *CVPR*, 2019. 2, 3, 6, 7

[37] Hangjie Yuan and Dong Ni. Learning visual context for group activity recognition. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Virtual Event, February 2-9, 2021*, pages 3261–3269. AAAI Press, 2021. 7

[38] Hangjie Yuan, Dong Ni, and Mang Wang. Spatio-temporal dynamic inference network for group activity recognition. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 7476–7485, 2021. 3, 7

[39] Chi Zhang, Baoxiong Jia, Mark Edmonds, Song-Chun Zhu, and Yixin Zhu. ACRE: abstract causal reasoning beyond covariation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 10643–10653. Computer Vision Foundation / IEEE, 2021. 3

[40] Peizhen Zhang, Yongyi Tang, Jianfang Hu, and Wei-Shi Zheng. Fast collective activity recognition under weak supervision. *IEEE Trans. Image Process.*, 29:29–43, 2020. 2