GroupFormer: Group Activity Recognition with Clustered Spatial-Temporal Transformer
(*Supplementary Materials*)

Shuaicheng Li[1*], Qianggang Cao[1*], Lingbo Liu[2], Kunlin Yang[1†], Shinan Liu[1], Jun Hou[1], Shuai Yi[1]
[1]Sensetime Research, [2]The Hong Kong Polytechnic University
{lishuaicheng,caoqianggang,yangkunlin,liushinan,houjun,yishuai}@sensetime.com
liulingbo918@gmail.com

In this supplementary material, Section 1 presents more details of our method . Section 2 describes the additional experiment settings. In Section 3, we conduct additional experiments to validate the effectiveness of different parts.

1. More Details of Method

For the clustered members, we compute two types of attention: (1) **intra-attention** as only queries and keys from the same cluster are taken into consideration. (2) **inter-attention** as a pairwise weighted connection of the clusters are taken into consideration.

In detail, we define a set of centroid vectors $\mathbf{M} = (m_1, ..., m_C) \in \mathbb{R}^{C\times D}$. It can be learned with the rest of the model parameters as implemented in [1]. We use k-means clustering algorithm to determine the cluster membership for these *actor queries* by Euclidean distance. Specifically, we sort queries by distance to every centroid. For each query, we compute euclidean distance between the query and every centroid, and assign the target query into the closest cluster. It is worth noting that different clusters may contain different numbers of queries.

Intra-attention: Formally, we denote the *actor query* as $Q \in \mathbb{R}^{N\times D}$, the i-th *actor query* denotes Q_i. Let us define $A \in \{0,1\}^{N\times C}$, which denotes a partition of the queries into C clusters. A_{ij} represent the i-th *actor query* Q_i belongs to the j-th cluster and 0 otherwise. Also, we define $Q^j = \{Q_i | A_{ij} = 1\}$, and it represents that j-th cluster contains corresponding query members. $|Q^j|$ denotes the number of j-th cluster's members. For the *actor query* belonging to j-th cluster, we define their candidate keys and values as K^j, V^j respectively shaped as $|Q^j| \times D$. Therefore, we update these actors' value as follows,

$$V'^j = \mathrm{softmax}(\frac{Q^j {K^j}^T}{\sqrt{D}})V^j \tag{1}$$

where j denotes j-th cluster and $V'^j \in \mathbb{R}^{|Q^j|\times D}$. Note that $\mathrm{softmax}(\cdot)$ is applied row-wise. We update the actors' values for every cluster and the final individual information can be represented as $V' \in \mathbb{R}^{N\times D}$.

Inter-attention: The intuitive tactic of building the relations of all clusters is to regard the clusters' centroid vectors as cluster holistic features $M \in \mathbb{R}^{C\times D}$. To capture the cluster inter-attention and update their cluster holistic information, we first embed the cluster feature into the query, key, and value. Then, the inter-attention is obtained by dot production operation and the row-wise $\mathrm{softmax}$. Using the attention weight, we compute the new values as the updated cluster holistic features. The process of above can be formulated as,

$$\hat{Q} = MW_q, \quad \hat{K} = MW_k, \quad \hat{V} = MW_v \tag{2}$$

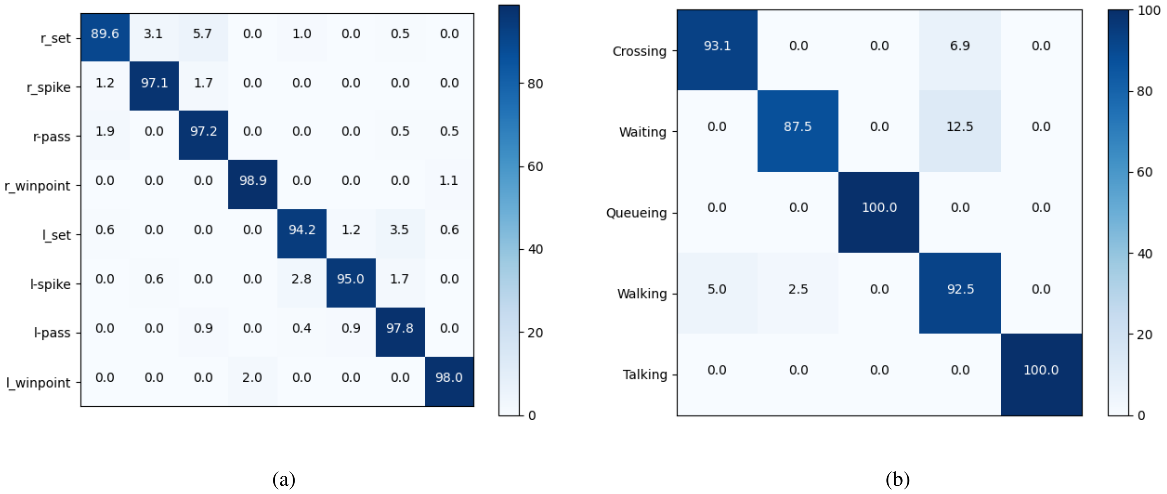
$$\hat{V}' = \mathrm{softmax}(\frac{\hat{Q}\hat{K}^T}{\sqrt{D}})\hat{V} \tag{3}$$

where the W_q, W_k, W_v represent the trainable parameter matrix shaped as $D \times D$.

Finally, we assign the holistic features into each queries belonging candidate cluster, which can be formulated as,

$$V_i^j = V_i'^j + \hat{V}'^j \tag{4}$$

where V_i^j denotes the values of the i-th actor query belonging to j-th cluster. $\hat{V}'^j$ denote the holistic features for j-cluster.



(a) (b)

Figure 1. We visualize the confusion matrix on both two datasets. (a) The left figure plots the confusion matrix on the Volleyball dataset and (b) the right figure displays the confusion matrix on the Collective dataset.

Types	Group Activity	Individual Action
GRG-none	93.4	83.2
GRG-actor	93.6	83.3
GRG-scene	93.8	83.5
Ours	**94.1**	**83.7**

Table 1. Performance comparison with various type of GRG.

2. Experiment Settings

Inception-v3 backbone: Apart from using the I3D backbone, we conduct experiments on the Inception-v3 backbone for fair comparisons with previous methods. We start by slicing out a T-frames ($T = 7$) centered the annotated clip, which denotes as $\mathbf{x_{img}} \in \mathbb{R}^{T\times 3\times H\times W}$ (with 3 color channels), and then adopt Inception-v3 [3] as the backbone to generate feature maps for each frame. We generate the feature maps at multiple resolutions. In practice, we extract the deep feature maps for each frame from the last convolutional layer denoted as $\mathbf{X_g} \in \mathbb{R}^{T\times C_g\times H'\times W'}$, which can be viewed as the scene feature for the entire video clip. In addition, we also generate the higher resolution feature maps $\mathbf{X_a}$ from the intermediate *Mixed_5d* layer followed [4] and then apply RoIAlign [2] to extract features for each individuals given N bounding boxes in each video frame. A fully connected layer is adopted to embed the aligned individual features into a D dimensional feature vector for each actor. Finally, we generate the individual representation $\mathbf{X} \in \mathbb{R}^{T\times N\times D}$.

3. Experiment

3.1. Confusion Matrix

To further understand the performance of our method, we draw the confusion matrix on the Volleyball dataset in Figure 1(a) and the Collective dataset in Figure 1(b) separately. For the Volleyball dataset, except for the "r_set", the other group activity accuracies achieve over 94%, indicating the effectiveness of our framework. More specifically, the main successful cases are from "winpoint" and "pass" within the left and right groups due to the effective semantic context learning. For the Collective dataset, our method achieves promising recognition accuracies on most of the activities such as "talking" and "queueing". Our model also struggles to distinguish between "waiting" and "walking", which is attributed to the temporally action evolution in our input video clips. In detail, "waiting" only describes the activity of one static frame in this clip, while it does not consider the temporal dynamic *e.g*. from "walking" to "waiting" in some video frames. Besides, the confusion emerges from discriminating "crossing" and "walking", which is probably due to similar actions in a video scenario.

3.2. Ablation Study on GRG

To validate the effectiveness of our Group Representation Generator (GRG), we design several variants including (1)*GRG-scene*: only extracts visual tokens from the holistic feature; (2)*GRG-actor*: only transforms the individual features into actor tokens; (3)*GRG-none*: discards GRG and introduces a learned query to be the initialized group representation. Results are shown in Table 1. Using our designed initialized group representation generator has a significant boost comparing with the other simpler generators. It indicates that both scene and individual information are beneficial to initializing group representation for the group activity inferring.

References

[1] Leon Bottou and Yoshua Bengio. Convergence properties of the k-means algorithms. In *Advances in neural information processing systems*, pages 585–592, 1995. 1

[2] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[3] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 2

[4] Jianchao Wu, Limin Wang, Li Wang, Jie Guo, and Gangshan Wu. Learning actor relation graphs for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9964–9974, 2019. 2