

COMPOSER: Compositional Reasoning of Group Activity in Videos with Keypoint-Only Modality

Honglu Zhou^{1*}, Asim Kadav², Aviv Shamsian³, Shijie Geng¹, Farley Lai²,
Long Zhao⁴, Ting Liu⁴, Mubbashir Kapadia¹, and Hans Peter Graf²

¹ Department of Computer Science, Rutgers University, Piscataway, NJ, USA

{hz289, sg1309, mk1353}@cs.rutgers.edu

² NEC Laboratories America, Inc., San Jose, CA, USA

{asim, farleylai, hpg}@nec-labs.com

³ Bar-Ilan University, Israel

aviv.shamsian@biu.ac.il

⁴ Google Research, Los Angeles, CA, USA

{longzh, liuti}@google.com

Abstract. Group Activity Recognition detects the activity collectively performed by a group of actors, which requires compositional reasoning of actors and objects. We approach the task by modeling the video as tokens that represent the multi-scale semantic concepts in the video. We propose COMPOSER, a Multiscale Transformer based architecture that performs attention-based *reasoning* over tokens at each scale and learns group activity *compositionally*. In addition, prior works suffer from scene biases with privacy and ethical concerns. We only use the keypoint modality which reduces scene biases and prevents acquiring detailed visual data that may contain private or biased information of users. We improve the multiscale representations in COMPOSER by clustering the intermediate scale representations, while maintaining consistent cluster assignments between scales. Finally, we use techniques such as auxiliary prediction and data augmentations tailored to the keypoint signals to aid model training. We demonstrate the model’s strength and interpretability on two widely-used datasets (Volleyball and Collective Activity). COMPOSER achieves up to +5.4% improvement with just the keypoint modality ¹.

Keywords: Keypoint-only group activity recognition · Compositional-ity · Multiscale representations · Transformer · Video understanding

1 Introduction

Group Activity Recognition (GAR) detects the activity collectively performed by a group of actors in a short video clip [19,108]. GAR has widespread societal implications in a variety of domains including security, surveillance, kinesiology, sports analysis, robot-human interaction, and rehabilitation [29,83,117,27].

* Work done as a NEC Labs intern.

¹ Code is available at <https://github.com/hongluzhou/composer>

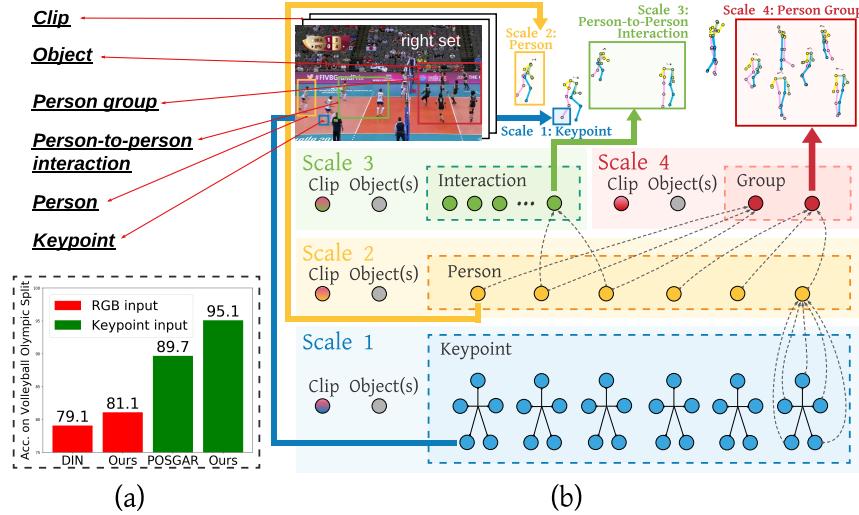


Fig. 1. (a) The keypoint-only setup generalizes better for GAR. The Volleyball Olympic split [102] ensures videos having vastly different scene background between training and testing, which can examine GAR model’s scene generalization ability. RGB-based methods severely suffer from scene biases and have poor model generalizability. **(b) Main idea.** We propose COMPOSER that uses keypoint only modality for GAR by modeling a video as *tokens* that represent the multiscale semantic concepts in the video, which include *keypoint*, *person*, person-to-person *interaction*, person *group*, *object* if present, and the *clip*. Four scales are formed by grouping actor-related tokens according to their semantic hierarchy. Representations of tokens in coarser scales are learned and aggregated from tokens of the finer scales. COMPOSER (Fig. 3) facilitates compositional reasoning of group activity in videos.

The task requires addressing two challenges. First, GAR requires a *compositional understanding* of the scene [3]. Because of the crowded scene, it is challenging to learn meaningful representations for GAR over the entire scene [108]. Since group activity often consists of sub-group(s) of actors and scene objects, the final action label depends on a compositional understanding of these entities [108,116]. Second, GAR benefits from *relational reasoning* over scene elements to understand the relative importance of entities and their interactions [41,113]. For example, in a volleyball game, persons around the ball performing the jumping action are more important than others standing in the scene.

Existing work has proposed to jointly learn the group activity with individual actions [47,90,84,45,8,5] or person sub-groups [65,75,27] for a compositional understanding of the group activity. Meanwhile, graph [117,46,107,41] and transformer [29,65] based models have been proposed for relational reasoning over scene entities. However, these methods do not sufficiently make use of the multiscale scene elements in the GAR task by modeling over entities at either one semantic scale (e.g., person [29,117,107,41]) or two scales (person and person

group [65,75,27], or keypoint and person [82]). More importantly, explicit multiscale modeling is neglected, lacking consistent compositional representations for the group action tasks. Furthermore, majority of the prior GAR methods rely on the RGB modality (see Table. 3), which causes the model more likely to have privacy and ethical issues when deployed in real-world applications [39]. Last but not least, the RGB input hinders the model’s robustness to changes in background, lighting conditions or textures, and often results in poor model generalizability due to scene biases (see Fig. 1 (a)) [17,94].

In this paper, we present **COMPOSER** that addresses *compositional learning* of entities in the video and *relational reasoning* about these entities. Inspired by how humans are particularly adept at representing objects in different granularities meanwhile reasoning their interactions to turn sensory signals into a high-level knowledge [43,60], we approach GAR by modeling a video as tokens that represent the multi-scale semantic concepts in the video (Fig. 1 (b)). Compared to the aforementioned prior works, we consider more fine-grained scene entities that are grouped into *four* scales. By combining the scales together with Multiscale Transformer (Fig. 4), **COMPOSER** provides attention-based reasoning over tokens at each scale, which makes the higher-level understanding of the group activity possible. Moreover, **COMPOSER** uses only the keypoint modality. Using only the 2D (or 3D) keypoints as input, our method can prevent the sensor camera from acquiring detailed visual data that may contain private or biased information of users². Keypoints also allow the model to focus on the action-specific cues, and help the model be more invariant to the scene biases. **COMPOSER** generalizes much better to testing data with different scene backgrounds (see the Volleyball Olympic split results in Table. 1).

COMPOSER learns *consistent* multiscale representations which boost the performance for GAR (Fig. 2). This is achieved by contrastive clustering assignments of clips. Intuitively, a model can recognize the group activity using representations of entities at just one particular scale. Hence, we consider representations of the clip token learned across scales as representations of different *views* of the clip. Such perspective allows us to cluster clip representations learned at all scales while enforcing consistency between cluster assignments produced from different scales of the *same* clip. In order to enforce this consistency, we follow [11] and use a *swapped* prediction mechanism where we predict the cluster assignment of a scale from the representation of another scale. However, distinct from related works [11,4,13], which use information from multiple augmentations or modalities for self-supervised learning from unlabelled images or videos, we use information from multiple scales for the task of group activity recognition. Contrasting clustering assignments enhance our intermediate representations and the overall performance. Finally, we use techniques such as auxiliary prediction at each scale and data augmentation methods such as *Actor Dropout* to aid training.

Our contributions are three-fold:

² Even for the keypoint extraction backbone which our method is agnostic to, there are existing works [39] that perform privacy-preserving keypoint estimation.

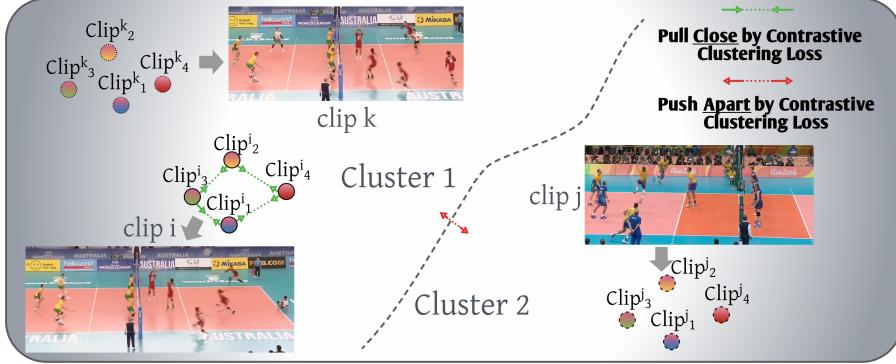


Fig. 2. Embedding space learned by COMPOSER. COMPOSER exploits a contrastive clustering objective (Sec. 3.3) to learn *consistent* multiscale representations for GAR. This is achieved by clustering clip representations learned at all scales. The clustering objective encourages an “agreement” between scales on the high-level knowledge learned (‘Pull Close’ representations of the same clip). Contrastive learning is performed on the clusters, which also helps the model to discriminate between clips with different semantic characteristics (‘Pull Close’ representations of the semantically-similar clips and ‘Push Apart’ those that are semantically-different). In the illustration, we use subscript to denote the scale and use superscript to indicate different clips.

1. We present COMPOSER for compositional reasoning of group activity in videos. COMPOSER can distill and convey high-level semantic knowledge from the elementary elements of the human-centered videos. We learn contrastive clustering assignment to improve the multiscale representations. By maintaining a consistent cluster assignment across the multiple scales of the *same* clip, an agreement between scales on the high-level knowledge learned can be promoted to optimize the representations across scales.
2. We use only the keypoint modality that allows COMPOSER to address the privacy and ethical concerns and to be robust to changes in background, with auxiliary prediction and data augmentation methods tailored to learning group activity from the keypoint modality.
3. We demonstrate the model’s strength and interpretability on two commonly-used datasets (Volleyball and Collective Activity) and COMPOSER achieves up to +5.4% improvement using just the keypoint modality.

2 Related Work

Much of the recent research on GAR explores how to capture the actor relations [46, 6, 107, 41, 83]. Several works tackle this problem from a graph-based perspective [46, 71, 113, 112]. Some utilize attention modeling [84, 109, 71, 116] including using Transformers [29, 65]. Existing works have primarily used RGB- and/or optical-flow-based features with ROIAlign [36] to represent actors [112, 84, 107, 8].

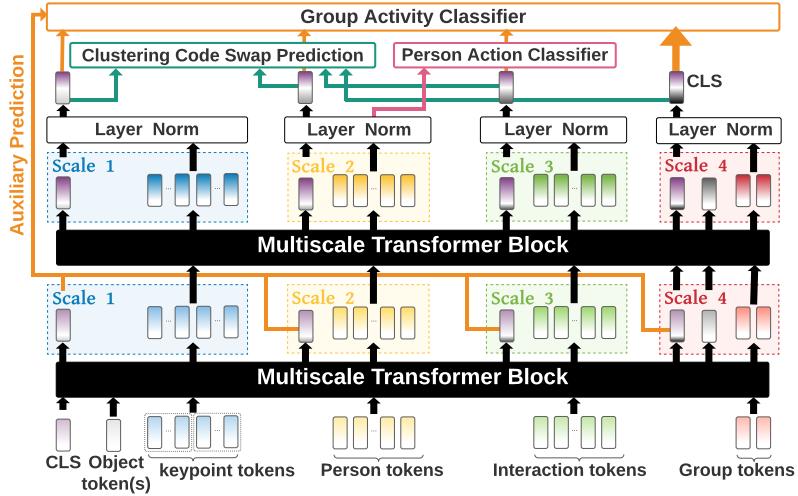


Fig. 3. COMPOSER. Given tokens that represent the multiscale semantic concepts (Fig. 1) in the human-centered video, COMPOSER jointly learns group activity, individual actions and contrastive clustering assignments of clips. Auxiliary predictions are enforced to aid training (Sec. 3.5).

A few recent works replace or augment these features with keypoints/poses of the actors [102, 82, 29, 116]. In this paper, we use only the light-weight coordinate-based keypoint representation. We propose a Multiscale Transformer block to hierarchically reason about entities at different semantic scales and we aid learning group activities by improving the musicale representations. Please see an in-depth discussion on related works in Appendix G.

3 Methodology

We present COMPOSER (Fig. 3), a novel Multiscale Transformer based architecture for GAR. In Sec. F.2, we describe the multi-scale semantic tokens representing a video with group activities. We introduce COMPOSER and especially its reasoning module Multiscale Transformer in Sec. 3.2. We describe data augmentations in Sec. 3.4 and the exact formulation of auxiliary prediction in Sec. 3.5.

3.1 Tokenizing a Video as Hierarchical Semantic Entities

We model a video as semantic tokens that allow our method easily adaptable to understanding any videos with multi-actor multi-object interactions [72].

- **Person Keypoint.** We define a person keypoint token, $\mathbf{k}_p^j \in \mathbb{R}^d$ that represents a keypoint joint j ($j = 1, \dots, j'$) of person p ($p = 1, \dots, p'$) in all timestamps, where j' is the number of joint types and p' is the number of actors. The initial d -dimensional person keypoint token is learned by encoding the numerical

coordinates (in the image space) of a certain keypoint track³. The procedure of encoding includes coordinate embedding, time positional embedding, keypoint type embedding, and OKS-based feature embedding [95] to mitigate the issue of noisy estimated keypoints. Details are available in Appendix F.2.

- **Person.** A person token is defined as $\mathbf{p}_p \in \mathbb{R}^d$, initially obtained by aggregating the standardized keypoint coordinates of person p over time through concatenation and FFN-based transformation.
- **Person-to-Person Interaction.** Modeling the person-to-person interactions is critical for GAR [108]. Unlike existing works that typically consider an interaction as an *edge* connecting two person nodes and learn a scalar to depict its importance [113], we model interaction as *nodes* (tokens) to allow for the modeling of complex higher-order interactions [72]. The person-to-person interaction token is defined as $\mathbf{i}_i \in \mathbb{R}^d$ where $i = 1, \dots, p' \times (p' - 1)$ (bi-directed interactions). Initial representation of the interaction between person p and q is learned from concatenation of \mathbf{p}_p and \mathbf{p}_q , followed by FFN-based transformation.
- **Person Group.** We define the group token $\mathbf{g}_g \in \mathbb{R}^d$ where $g = 1, \dots, g'$ for videos where sub-groups are often separable. g' denotes the num. of sub-groups in the video. Given the person-to-group mapping which can be obtained through various mechanisms (e.g., heuristics [82], k-means [65], etc [27,58]), representation of a group is an aggregate over representations of persons in the group similarly through concatenation and FFN.
- **Clip.** The special [CLS] token ($\in \mathbb{R}^d$) is a learnable embedding vector and is considered as the clip representation. CLS stands for classification and is often used in Transformers to “summarize” the task-related representative information from all tokens in the input sequence [26].
- **Object.** Scene objects can play a crucial role in videos where human(s) interact with object(s). E.g., in a volleyball game where one person is spiking and multiple nearby actors are all jumping with arms up, it can be difficult to tell which person is the key person with information of just the person keypoints due to their similar poses. The ball keypoints can help to distinguish the key person. Object keypoints can be used to represent an object in the scene with similar benefits of person keypoints (e.g., to boost model robustness [49]). Object keypoint detection [9,69] benefits downstream tasks such as human action recognition [42], object detection [49,115], tracking [76], etc [59]. Thus, we use object keypoints to represent each object for GAR. We denote object token $\mathbf{e}_e \in \mathbb{R}^d$ where $e = 1, \dots, e'$ and e' is the maximal number of objects a video might have. Similar to person tokens, the initial object tokens are learned from aggregating the coordinate-represented object keypoints.

3.2 Multiscale Transformer

Multiscale Transformer takes a sequence of multiple-scale tokens as input, and refines representations of these tokens. Specifically, tokens of the four scales are:

³ We use track-based representations [102,119,29,65] to represent each token.

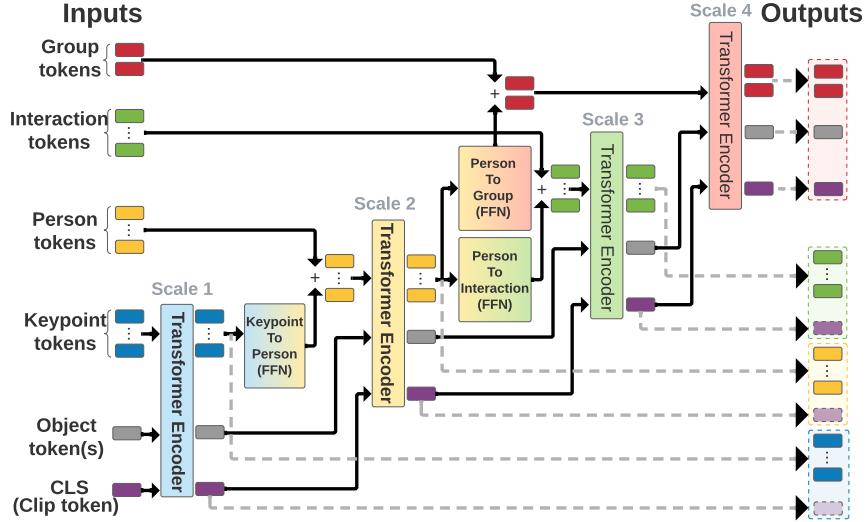


Fig. 4. Multiscale Transformer performs relational reasoning with four Transformer Encoders to operate self-attention on tokens of each scale, while stringing tokens of the four scales together with FFNs and Skip Connections to learn hierarchical representations that make a high-level understanding of group activity possible.

$$\begin{aligned}
 \text{Scale 1: } & \left\{ [\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{k}_1^1, \dots, \mathbf{k}_{p'}^{j'} \right\}, \\
 \text{Scale 2: } & \left\{ [\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{p}_1, \dots, \mathbf{p}_{p'} \right\}, \\
 \text{Scale 3: } & \left\{ [\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{i}_1, \dots, \mathbf{i}_{p' \times (p'-1)} \right\}, \\
 \text{Scale 4: } & \left\{ [\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{g}_1, \dots, \mathbf{g}_{g'} \right\}.
 \end{aligned} \tag{1}$$

We utilize a Transformer encoder [103] at each scale to perform relational reasoning of tokens in that scale. We review details of Transformer in Appendix F.1.

Hierarchical representations of tokens are maintained in an elaborately designed Multiscale Transformer block (Fig. 4). In the Multiscale Transformer block, operations in the four scales are the same (but with different parameters) to maintain simplicity. Specifically, given a sequence of tokens of scale s (Eq. 1), Transformer encoder outputs refined representations of these tokens. Then, concatenation and FFN are used to aggregate refined representations of *actor-related* tokens, in order to form representations of actor-related tokens in the subsequent coarser scale $s+1$. Such learned representations are summed with their initial representations (input to the Multiscale Transformer) (i.e. Skip Connection). The resulting actor-related tokens, as well as scale s updated [CLS] token and object token(s) form the input sequence of the Transformer encoder in the scale $s+1$ (see wiring in Fig. 4).

COMPOSER uses the initial representations of the multi-scale semantic tokens (Sec. F.2) as input, and utilizes multiple blocks of Multiscale Transformer to

perform relational reasoning over these tokens. With refined token representations, COMPOSER *jointly* learns group activity, individual actions and contrastive clustering of clips (the multitask-learning details are in Sec. 3.5).

3.3 Contrastive Clustering for Scale Agreement

We consider the clip tokens learned at different scales as representations of different *views* of the clip instance. Then, we cluster clip representations learned in all scales while enforcing consistency between cluster assignments produced from different scales of the clip. This can act as regularization of the embedding space during training (Fig. 2). To enforce consistency, we use a swapped prediction mechanism [11] where we predict the cluster assignment of a scale from the representation of another scale. COMPOSER jointly learns GAR and the swapped prediction task to capture an agreement of the common semantic information hidden across the scales.

Preliminaries. Suppose $\mathbf{v}_{n,s} \in \mathbb{R}^d$ represents the learned representation of clip n in scale s , where $s \in \{1, 2, 3, 4\}$. Following prior works [11, 52], we first project the representation to the unit sphere. We then compute a code (i.e., cluster assignment) $\mathbf{q}_{n,s} \in \mathbb{R}^K$ by mapping $\mathbf{v}_{n,s}$ to a set of K *trainable* prototype vectors, $\{\mathbf{c}_1, \dots, \mathbf{c}_K\}$. We denote by $C \in \mathbb{R}^{K \times d}$ the matrix whose rows are the $\mathbf{c}_1, \dots, \mathbf{c}_K$.

Swapped Prediction. Suppose s and w denote 2 different scales from the four representation scales. The swapped prediction problem aims to predict the code $\mathbf{q}_{n,s}$ from $\mathbf{v}_{n,w}$, and $\mathbf{q}_{n,w}$ from $\mathbf{v}_{n,s}$, with the following loss function:

$$\mathcal{L}_{\text{swap}}(\mathbf{v}_{n,w}, \mathbf{v}_{n,s}) = \ell(\mathbf{v}_{n,w}, \mathbf{q}_{n,s}) + \ell(\mathbf{v}_{n,s}, \mathbf{q}_{n,w}) \quad (2)$$

where $\ell(\mathbf{v}_{n,w}, \mathbf{q}_{n,s})$ measures the fit between $\mathbf{v}_{n,w}$ and $\mathbf{q}_{n,s}$. $\ell(\mathbf{v}_{n,w}, \mathbf{q}_{n,s})$ is the cross entropy loss between $\mathbf{q}_{n,s}$ and the probability obtained by taking a softmax of the dot products of $\mathbf{v}_{n,w}$ and prototypes in C :

$$\ell(\mathbf{v}_{n,w}, \mathbf{q}_{n,s}) = - \sum_{k=1}^K \mathbf{q}_{n,s}^{(k)} \log \frac{\exp(\frac{1}{\tau} \mathbf{v}_{n,w} \mathbf{c}_k^\top)}{\sum_{k'=1}^K \exp(\frac{1}{\tau} \mathbf{v}_{n,w} \mathbf{c}_{k'}^\top)} \quad (3)$$

where τ is a temperature parameter. The total loss of the swapped prediction problem is taking Eq. (2) computed over all pairs of scales and all N clips,

$$\mathcal{L}_{\text{cluster}} = \frac{1}{N} \sum_{n=1}^N \left(\sum_{w,s \in \{1,2,3,4\} \& w \neq s} \mathcal{L}_{\text{swap}}(\mathbf{v}_{n,w}, \mathbf{v}_{n,s}) \right) \quad (4)$$

Online Clustering. This step produces the cluster assignments using the learned prototypes C and the learned clip representations *only within a batch*, $V \in \mathbb{R}^{B \times d}$ where B denotes the batch size. We perform the clustering in an online fashion for faster training and use the method proposed in [11]. Specifically, online clustering yields the codes $Q \in \mathbb{R}^{B \times K}$. We compute codes Q such that all examples in a batch are equally partitioned by the prototypes (which prevents the trivial

solution where every clip has the same code). Q is optimized to maximize the similarity between the learned clip representations and the prototypes,

$$\max_{Q \in \mathcal{Q}} \text{Tr}(QCV^\top) + \varepsilon H(Q), \quad (5)$$

$$\mathcal{Q} = \left\{ Q \in \mathbb{R}_+^{B \times K} \mid \mathbf{1}_B Q = \frac{1}{K} \mathbf{1}_K, Q \mathbf{1}_K^\top = \frac{1}{B} \mathbf{1}_B^\top \right\}$$

where the trace Tr is the sum of the elements on the main diagonal, H is the entropy function, and ε is a parameter that controls the smoothness of the mapping. $\mathbf{1}_K \in \mathbb{R}^K$ and $\mathbf{1}_B \in \mathbb{R}^B$ are a vector of ones to enforce the equipartition constraint. The continuous solution Q^* of Eq. (5) is computed with the iterative Sinkhorn-Knopp algorithm [22,11].

3.4 Data Augmentation for Keypoint Modality

We use the following data augmentations to aid training and improve generalization ability of the model learned from the *keypoint* modality.

Actor Dropout is performed by removing a random actor in a random frame, inspired by [77] that masks agents with probabilities to predict agent behaviors for autonomous driving. We remove actors by replacing the representation of the actor with a zero vector.

Horizontal Flip is often used by existing GAR methods [119,102,82], which is performed on the video frame level. This augmentation causes the pose of each person and positions of (left and right) sub-groups flipped horizontally. We add a small random perturbation on each flipped keypoint.

Horizontal Move means we horizontally move all keypoints in the clip by a certain number of pixel locations, which is randomly determined per video and bounded by a pre-defined number (i.e., 10). Similarly, afterwards a small random perturbation is applied on each keypoint.

Vertical Move is done similar to the Horizontal Move, except we move the keypoints in the vertical direction.

Novel practices like Actor Dropout, Horizontal/Vertical Move and random perturbations help the model to perform GAR from noisy estimated keypoints.

3.5 Auxiliary Prediction and Multitask Learning

We take the learned representation of the clip at *each* scale of *each* Multiscale Transformer block, and perform *auxiliary group activity predictions* (Fig. 3). Specifically, each of the clip representations learned at each scale of each block is sent as input to the group activity classifier to produce one GAR result. In addition, person representation from the last Multiscale Transformer block is the input to a person action classifier. Meanwhile, the loss of the swapped prediction problem is computed given the learned representations of the clip of all 4 scales from the last Multiscale Transformer block. The total loss is:

$$\mathcal{L}_{\text{total}} = \sum_{m=1}^{M-1} \mathcal{L}_{\text{groupAux}} + \lambda (\mathcal{L}_{\text{groupLast}} + \mathcal{L}_{\text{person}} + \mathcal{L}_{\text{cluster}}) \quad (6)$$

where $\mathcal{L}_{\text{groupAux}}$ represents the loss from Auxiliary Prediction incurred by clip representations at different scales and early blocks of the Multiscale Transformer, $\mathcal{L}_{\text{groupLast}}$ is from the last Multiscale Transformer block, $\mathcal{L}_{\text{person}}$ is the person action classification loss, and $\mathcal{L}_{\text{cluster}}$ is the contrastive clustering loss (Eq. 4). m denotes the index of the Multiscale Transformer block, M is the total number of the Multiscale Transformer blocks, and λ is a hyper-parameter that weights the importance of predictions from the last block. For metric evaluation, we use the clip token from the last scale in the last Multiscale Transformer as input to the group activity classifier.

4 Experimental Evaluation

4.1 Dataset

The Volleyball dataset [47] (VD) comprises 4,830 clips from 55 videos. The group activity labels include 8 activities: 4 main activities (*set, spike, pass, win-point*) which are divided into two subgroups, *left* and *right*. Each player can perform one of the 9 actions: *blocking, digging, falling, jumping, moving, setting, spiking, standing* and *waiting*. The dataset has a default ‘Original’ split in which train/test videos were randomly splitted (39 train and 16 test videos). A skewed ‘Olympic’ split [102] was later released in which train/test videos are splitted according to the match venues: 29 train videos are from the same 2012 London Olympics venue, while the rest 26 test videos are from numerous venues, and thus largely differs from the train videos w.r.t. the scene background.

The Collective Activity dataset [20] (CAD) is a dataset with 44 real-life videos [108]. The group activity labels are *crossing, waiting, queueing, walking* and *talking* (person action labels have an additional ‘N/A’ class). We follow prior works to merge the class *crossing* and *walking* into *moving* [117,106,110,112], and use the same train-test split [117,107,84] and actor tracklets [117,8]. Please refer to Appendix F for implementation details on both datasets.

4.2 Comparison with State-of-the-Arts

Scene Generalization for Keypoint-only Setup To support the keypoint-only setup for GAR, we first compare the generalization capability of models using either RGB or the keypoint modality. In Table 1, I3D and VGG-16 are two commonly-used image backbone by prior RGB-based GAR methods; the rest are all GAR models (all use VGG-16 as the backbone).

On VD Olympic split, the best prior RGB-based method is DIN [117] in Table 1. We substitute DIN with a COMPOSER variant ⁴ (Sec. 1) that also consumes RGB input instead of keypoint, and the result is 81.1% which is 2% higher than DIN, suggesting the stronger reasoning strength of COMPOSER, but the accuracy is still low due to the RGB signals. POGARS [102] uses the keypoint

⁴ This COMPOSER variant consumes RGB-based ROI-aligned person features as input, and thus only models 3 scales: person, interaction, and the group scale.

Table 1. Test accuracy on VD under different train/test splits. Yellow shaded rows highlight the methods use RGB input, and blue for keypoint

Model	VD Acc. (%) ↑	
	Olympic	Original
I3D [12]	73.9	84.6
VGG-16 [93]	76.4	91.6
PCTDM [110]	75.2	91.7
SACRF [83]	71.1	91.8
AT [29]	76.9	93.0
ARG [107]	77.8	93.3
TCE-STBiP [116]	78.5	93.5
DIN [117]	79.1	93.6
POGARS [102]	89.7	93.2
COMPOSER (ours)	95.1	93.7
Improvement	+5.4%	+0.1%

*Note: Keypoint-based methods do NOT use ball keypoint in this table in order to have a rigorous comparison because RGB-based methods are unaware of such info.

Table 2. Comparisons with state-of-the-art (SOTA) methods that leverage **only keypoint information** on the **VD Original split**. **COMPOSER** outperforms existing methods and achieves a new highest record (+0.7% improvement)

Model	Keypoint		Acc.
	Actor	Object	
Zappardino <i>et al.</i> [119]	✓		91.0
GIRN [82]	✓		88.4
	✓	✓	92.2
AT [29]	✓		92.3
	✓	✓	92.8
POGARS [102]	✓		93.2
	✓	✓	93.9
COMPOSER (ours)	✓		93.7
	✓	✓	94.6

modality and has an accuracy of 89.7%, higher than all RGB-based methods. **COMPOSER** with the keypoint-only modality obtains 95.1% accuracy and *significantly outperforms* prior methods, yielding +5.4% improvement. These results imply that the keypoint-only setup can reduce scene biases, and generalize better than approaches relying on the RGB modality to testing data with different visual characteristics from training.

We also report the results of these methods that we obtained on VD Original split in Table 1. From this side-by-side comparison, the difference between the Olympic and Original split is vivid. Current GAR methods have quite saturated performances on the Original split of VD and the results are all very high (more evidence later). Therefore, we recommend readers using the more challenging VD Olympic split for future research on GAR. Note that the **COMPOSER** that outperforms prior methods in Table 1 is only an ablated version of ours in that not using the object token(s). In addition, GroupFormer [65] is currently the best-performing method (Table 4 in Appendix) and its RGB-only variant has the result of 94.1% accuracy on VD Original split. However, GroupFormer uses additional scene features with the Inception-v3 backbone.

Comparisons of Methods Using Keypoint-only Modality In Table 2, we compare **COMPOSER** with more GAR methods that use only the keypoint modality on VD Original split following conventions. **COMPOSER** achieves a new SOTA 94.6% accuracy with +0.7% improvement.

Table 3. Comparisons with SOTA methods that use a single or multiple modalities on the original split of VD and CAD. ‘‘Flow’’ denotes optical flow input, and ‘‘Scene’’ denotes features of the entire frames. Fewer modalities indicates a stronger capability of the model itself (*fewer checks are better*). The top 3 performance scores are highlighted as: **First**, **Second***, **Third**. COMPOSER outperforms the latest GAR methods that use a single modality (+0.7% improvement on VD and +2.8% improvement on CAD), and performs favorably compared with methods that exploit multiple expensive modalities

Model	Modality				Dataset	
	Keypoint	RGB	Flow	Scene	VD	CAD
HDTM [47]		✓			81.9	81.5
CERN [90]		✓			83.3	87.2
stagNet [84]		✓			89.3	89.1
RCRG [45]		✓			89.5	N/A
SSU [8]		✓			90.6	N/A
PRL [41]		✓			91.4	N/A
ARG [107]		✓			92.5	91.0
HiGCIN [112]		✓			91.5	93.4
DIN [117]		✓			93.6	N/A
Zappardino <i>et al.</i> [119]	✓				91.0	N/A
GIRN [82]	✓				92.2	N/A
AT [29]	✓				92.3	N/A
POGARS [102]	✓				93.9	N/A
CRM [5]		✓	✓		93.0	85.8
AT [29]		✓	✓		93.0	92.8
Ehsanpour <i>et al.</i> [27]		✓		✓	93.1	89.4
GIRN [82]	✓	✓	✓		94.0	N/A
TCE+STBiP [116]	✓	✓		✓	94.7	N/A
SACRF [83]	✓	✓	✓	✓	95.0*	95.2
GroupFormer [65]	✓	✓	✓	✓	95.7	96.3
COMPOSER (ours)	✓				94.6	96.2*

*Note: The best results of each method that were reported by the method authors are listed in the table in order to be compared with ours most rigidly. ‘‘N/A’’ stands for ‘not available’. Yellow shaded rows highlight that the methods use just the RGB-based input, whereas blue for just keypoint.

Among these methods, Zappardino *et al.* [119] use CNNs to learn group activity in Volleyball games, given sequence of person keypoint coordinates, their temporal differences, and keypoint differences from each actor to the pivot-actor that is selected by the model. The model does not model human-object interactions. AT [29] does not consider human-object interactions either, but because AT is also a Transformer-based model like ours, we can easily improve it by feeding our object tokens as additional inputs to AT. Moreover, GIRN [82] and POGARS [102] are designed to leverage ball trajectory for learning group activity in videos of Volleyball games. As shown in Table 2, the object keypoint information can greatly boost the performance by providing additional context. GIRN models interactions between joints within an actor and across actors, as well as joint-object interactions. POGARS uses 1D CNNs to learn spatiotemporal dynamics of actors. AT, GIRN, and POGARS all use dot-product-based attention mechanisms similar to ours, however, they fail to fully model the hier-

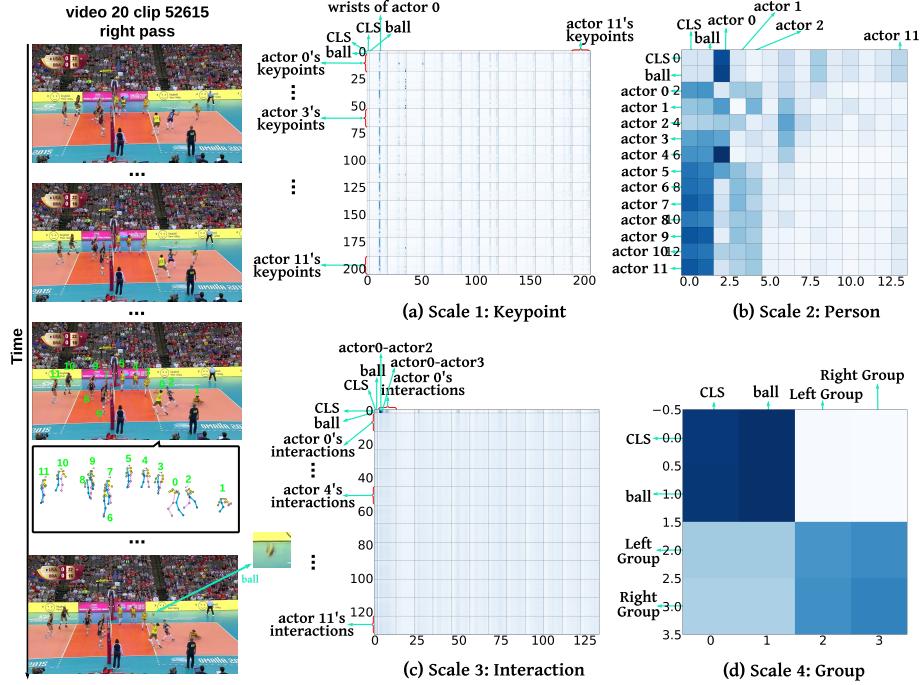


Fig. 5. Qualitative results of COMPOSER on VD – showcasing attention matrices of an instance in the “right pass” class (key actor is actor 0).

archical entities in the video (e.g., they all only use attention to learn person-wise importance, and at most consider two scales: keypoint and person), and more importantly, they lack explicit strategy to improve the multiscale representations in order to aid the compositional reasoning of group activity recognition.

Comparisons of Methods Using Other Modalities We compare results of COMPOSER with the best reported results of SOTA methods that use a single or multiple modalities in Table 3 on both VD and CAD. COMPOSER still achieves competitive performance – outperforming methods that use only RGB signals, obtaining +0.7% improvement on VD and +2.8% improvement on CAD if compared with methods that use a single modality (RGB or keypoint), and performing favorably compared with methods that exploit multiple expensive input modalities.

GroupFormer [65] has the highest accuracy on VD and CAD due to learning the representations of the multiscale scene entities (person and person group) with a Clustered Spatial-Temporal Transformer, and leveraging scene context and multiple *expensive* modalities (FLOPs: GroupFormer **595M** v.s. COMPOSER **297M**; details are in Appendix C).



Fig. 6. Qualitative results on CAD (video ID ‘10’). COMPOSER successfully predicts ‘Queueing’ even when the input keypoints are partially noisy due to occlusion.

4.3 Qualitative Results

We visualize the attention weights in Fig. 5. We highlight the tokens that the model has mostly attended to at each scale (e.g., wrists of actor 0 at the person keypoint scale). COMPOSER is able to attend to relevant information across different scales, and it can produce interpretable results. In Fig. 6, we visualize the keypoint input to COMPOSER on a CAD instance. COMPOSER implicitly learns the human motion patterns from the keypoint features to handle partial occlusions.

Please check Appendix for more analyses including ablation studies, confusion matrices, parameter sensitivity analyses w.r.t. the number of scales and the number of prototypes, more qualitative results including failure cases, etc.

5 Conclusion

We propose COMPOSER that uses a Multiscale Transformer to learn compositional reasoning at different scales for group activity recognition. We also improve the intermediate representations using contrastive clustering, auxiliary prediction, and data augmentation techniques. We demonstrate the model’s strength and interpretability on two widely-used datasets (Volleyball and Collective Activity). COMPOSER achieves up to +5.4% improvement with just the keypoint modality.

One limitation is that videos with severe occlusions remain challenging for COMPOSER like other existing methods, due to errors from detecting keypoints. Adopting 3D keypoints or stronger backbones that estimate keypoints directly from the video [80,2] can help to address the issue. Possible future directions include 1) expanding our methods to more complex scenarios, such as crowd understanding that may require modeling additional hierarchical scales; and 2) exploring effective multimodal fusion methods in order to use additional modalities like RGB but without suffering from scene biases, since RGB can be beneficial for activities that involve significant interaction with the background scene.

Acknowledgments The research was supported in part by NSF awards: IIS-1703883, IIS-1955404, IIS-1955365, RETTL-2119265, and EAGER-2122119. This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 22STESE00001 01 01. Disclaimer: The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

References

1. Ball recognition and tracking in live volleyball game. <https://github.com/tprlab/vball>
2. Mediapipe pose: ML solution for high-fidelity body pose tracking from rgb video frames. <https://google.github.io/mediapipe/solutions/pose.html>
3. Abkenar, A.B., Loke, S.W., Zaslavsky, A., Rahayu, W.: Groupsense: recognizing and understanding group physical activities using multi-device embedded sensing. ACM Transactions on Embedded Computing Systems (TECS) **17**(6), 1–26 (2019)
4. Asano, Y.M., Patrick, M., Rupprecht, C., Vedaldi, A.: Labelling unlabelled videos from scratch with multi-modal self-supervision. arXiv preprint arXiv:2006.13662 (2020)
5. Azar, S.M., Atigh, M.G., Nickabadi, A., Alahi, A.: Convolutional relational machine for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7892–7901 (2019)
6. Azar, S.M., Atigh, M.G., Nickabadi, A., Alahi, A.: Convolutional relational machine for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7892–7901 (2019)
7. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
8. Bagautdinov, T., Alahi, A., Fleuret, F., Fua, P., Savarese, S.: Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4315–4324 (2017)
9. Blomqvist, K., Chung, J.J., Ott, L., Siegwart, R.: Semi-automatic 3d object key-point annotation and detection for the masses. arXiv preprint arXiv:2201.07665 (2022)
10. Bottou, L.: From machine learning to machine reasoning. Machine learning **94**(2), 133–149 (2014)
11. Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A.: Unsupervised learning of visual features by contrasting cluster assignments. In: Thirty-fourth Conference on Neural Information Processing Systems (NeurIPS) (2020)
12. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
13. Chen, B., Rouditchenko, A., Duarte, K., Kuehne, H., Thomas, S., Boggust, A., Panda, R., Kingsbury, B., Feris, R., Harwath, D., et al.: Multimodal clustering networks for self-supervised learning from unlabeled videos. arXiv preprint arXiv:2104.12671 (2021)
14. Chen, H.Y., Lai, S.H.: Group activity recognition via computing human pose motion history and collective map from video. In: Asian Conference on Pattern Recognition. pp. 705–718. Springer (2019)
15. Chen, J., Hao, H., Hong, H., Kong, Y.: Rit-18: A novel dataset for compositional group activity understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 362–363 (2020)
16. Cheng, Z., Qin, L., Huang, Q., Jiang, S., Tian, Q.: Group activity recognition by gaussian processes estimation. In: 2010 20th International Conference on Pattern Recognition. pp. 3228–3231. IEEE (2010)
17. Choi, J., Gao, C., Messou, J.C., Huang, J.B.: Why can't i dance in a mall? learning to mitigate scene bias in action recognition. In: Proceedings of the 33rd

- International Conference on Neural Information Processing Systems. pp. 853–865 (2019)
18. Choi, W., Savarese, S.: A unified framework for multi-target tracking and collective activity recognition. In: European Conference on Computer Vision. pp. 215–230. Springer (2012)
 19. Choi, W., Savarese, S.: Understanding collective activities of people from videos. *IEEE transactions on pattern analysis and machine intelligence* **36**(6), 1242–1257 (2013)
 20. Choi, W., Shahid, K., Savarese, S.: What are they doing?: Collective activity classification using spatio-temporal relationship among people. In: 2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops. pp. 1282–1289. IEEE (2009)
 21. Choi, W., Shahid, K., Savarese, S.: Learning context for collective activity recognition. In: CVPR 2011. pp. 3273–3280. IEEE (2011)
 22. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* **26**, 2292–2300 (2013)
 23. Dang, L.M., Min, K., Wang, H., Piran, M.J., Lee, C.H., Moon, H.: Sensor-based and vision-based human activity recognition: A comprehensive survey. *Pattern Recognition* **108**, 107561 (2020)
 24. Dankers, V., Bruni, E., Hupkes, D.: The paradox of the compositionality of natural language: a neural machine translation case study. arXiv preprint arXiv:2108.05885 (2021)
 25. Deng, Z., Vahdat, A., Hu, H., Mori, G.: Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4772–4781 (2016)
 26. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
 27. Ehsanpour, M., Abedin, A., Saleh, F., Shi, J., Reid, I., Rezatofighi, H.: Joint learning of social groups, individuals action and sub-group activities in videos. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16. pp. 177–195. Springer (2020)
 28. Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. arXiv preprint arXiv:2104.11227 (2021)
 29. Gavriluk, K., Sanford, R., Javan, M., Snoek, C.G.: Actor-transformers for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 839–848 (2020)
 30. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 315–323. JMLR Workshop and Conference Proceedings (2011)
 31. Gong, Z., Zhong, P., Yu, Y., Hu, W., Li, S.: A cnn with multiscale convolution and diversified metric for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* **57**(6), 3599–3618 (2019)
 32. Grunde-McLaughlin, M., Krishna, R., Agrawala, M.: Agqa: A benchmark for compositional spatio-temporal reasoning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11287–11297 (2021)
 33. Haber, E., Ruthotto, L., Holtham, E., Jun, S.H.: Learning across scales—multiscale methods for convolution neural networks. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)

34. Hajimirsadeghi, H., Yan, W., Vahdat, A., Mori, G.: Visual recognition by counting instances: A multi-instance cardinality potential kernel. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2596–2605 (2015)
35. Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., Wang, Y.: Transformer in transformer. arXiv preprint arXiv:2103.00112 (2021)
36. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
37. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
38. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
39. Hinojosa, C., Niebles, J.C., Arguello, H.: Learning privacy-preserving optics for human pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2573–2582 (2021)
40. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
41. Hu, G., Cui, B., He, Y., Yu, S.: Progressive relation learning for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 980–989 (2020)
42. Huang, Y., Kadav, A., Lai, F., Patel, D., Graf, H.P.: Learning higher-order object interactions for keypoint-based video understanding (2021)
43. Hudson, D., Manning, C.D.: Learning by abstraction: The neural state machine. Advances in Neural Information Processing Systems **32**, 5903–5916 (2019)
44. Hudson, D.A., Manning, C.D.: Compositional attention networks for machine reasoning. In: International Conference on Learning Representations (2018)
45. Ibrahim, M.S., Mori, G.: Hierarchical relational networks for group activity recognition and retrieval. In: Proceedings of the European conference on computer vision (ECCV). pp. 721–736 (2018)
46. Ibrahim, M.S., Mori, G.: Hierarchical relational networks for group activity recognition and retrieval. In: Proceedings of the European conference on computer vision (ECCV). pp. 721–736 (2018)
47. Ibrahim, M.S., Muralidharan, S., Deng, Z., Vahdat, A., Mori, G.: A hierarchical deep temporal model for group activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1971–1980 (2016)
48. Irsay, O., Cardie, C.: Deep recursive neural networks for compositionality in language. In: Advances in neural information processing systems. pp. 2096–2104 (2014)
49. Jaiswal, A., Singh, S., Wu, Y., Natarajan, P., Natarajan, P.: Keypoints-aware object detection. In: NeurIPS 2020 Workshop on Pre-registration in Machine Learning. pp. 62–72. PMLR (2021)
50. Ji, J., Krishna, R., Fei-Fei, L., Niebles, J.C.: Action genome: Actions as compositions of spatio-temporal scene graphs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10236–10247 (2020)
51. Jia, B., Chen, Y., Huang, S., Zhu, Y., Zhu, S.c.: Lemma: A multi-view dataset for learning multi-agent multi-task activities. In: European Conference on Computer Vision. pp. 767–786. Springer (2020)
52. Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., Krishnan, D.: Supervised contrastive learning. arXiv preprint arXiv:2004.11362 (2020)

53. Kim, P.S., Lee, D.G., Lee, S.W.: Discriminative context learning with gated recurrent unit for group activity recognition. *Pattern Recognition* **76**, 149–161 (2018)
54. Kim, T.S., Hager, G.D.: Safcar: Structured attention fusion for compositional action recognition. arXiv preprint arXiv:2012.02109 (2020)
55. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
56. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
57. Kong, Y., Jia, Y., Fu, Y.: Learning human interaction by interactive phrases. In: European conference on computer vision. pp. 300–313. Springer (2012)
58. Koshkina, M., Pidaparthi, H., Elder, J.H.: Contrastive learning for sports video: Unsupervised player classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4528–4536 (2021)
59. Kulkarni, T.D., Gupta, A., Ionescu, C., Borgeaud, S., Reynolds, M., Zisserman, A., Mnih, V.: Unsupervised learning of object keypoints for perception and control. *Advances in neural information processing systems* **32** (2019)
60. Lake, B.M., Ullman, T.D., Tenenbaum, J.B., Gershman, S.J.: Building machines that learn and think like people. *Behavioral and brain sciences* **40** (2017)
61. Lan, T., Wang, Y., Yang, W., Robinovitch, S.N., Mori, G.: Discriminative latent models for recognizing contextual group activities. *IEEE transactions on pattern analysis and machine intelligence* **34**(8), 1549–1562 (2011)
62. Le, H., Chen, N.F., Hoi, S.C.: C³: Compositional counterfactual contrastive learning for video-grounded dialogues. arXiv preprint arXiv:2106.08914 (2021)
63. Li, G., Yu, Y.: Visual saliency detection based on multiscale deep cnn features. *IEEE transactions on image processing* **25**(11), 5012–5024 (2016)
64. Li, M., Chen, S., Zhao, Y., Zhang, Y., Wang, Y., Tian, Q.: Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 214–223 (2020)
65. Li, S., Cao, Q., Liu, L., Yang, K., Liu, S., Hou, J., Yi, S.: Groupformer: Group activity recognition with clustered spatial-temporal transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13668–13677 (2021)
66. Li, X., Choo Chuah, M.: Sbgar: Semantics based group activity recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 2876–2885 (2017)
67. Lin, L., Song, S., Yang, W., Liu, J.: Ms2l: Multi-task self-supervised learning for skeleton based action recognition. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 2490–2498 (2020)
68. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)
69. Lu, C., Koniusz, P.: Few-shot keypoint detection with uncertainty learning for unseen species. arXiv preprint arXiv:2112.06183 (2021)
70. Lu, L., Di, H., Lu, Y., Zhang, L., Wang, S.: Spatio-temporal attention mechanisms based model for collective activity recognition. *Signal Processing: Image Communication* **74**, 162–174 (2019)
71. Lu, L., Lu, Y., Yu, R., Di, H., Zhang, L., Wang, S.: Gaim: Graph attention interaction model for collective activity recognition. *IEEE Transactions on Multimedia* **22**(2), 524–539 (2019)

72. Luo, Z., Xie, W., Kapoor, S., Liang, Y., Cooper, M., Niebles, J.C., Adeli, E., Li, F.F.: Moma: Multi-object multi-actor activity parsing. *Advances in Neural Information Processing Systems* **34** (2021)
73. Materzynska, J., Xiao, T., Herzig, R., Xu, H., Wang, X., Darrell, T.: Something-else: Compositional action recognition with spatial-temporal interaction networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 1049–1059 (2020)
74. Nabi, M., Bue, A., Murino, V.: Temporal poselets for collective activity detection and recognition. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*. pp. 500–507 (2013)
75. Nakatani, C., Sendo, K., Ukita, N.: Group activity recognition using joint learning of individual action recognition and people grouping. In: *2021 17th International Conference on Machine Vision and Applications (MVA)*. pp. 1–5. IEEE (2021)
76. Nebehay, G., Pflugfelder, R.: Consensus-based matching and tracking of keypoints for object tracking. In: *IEEE Winter Conference on Applications of Computer Vision*. pp. 862–869. IEEE (2014)
77. Ngiam, J., Caine, B., Vasudevan, V., Zhang, Z., Chiang, H.T.L., Ling, J., Roelofs, R., Bewley, A., Liu, C., Venugopal, A., et al.: Scene transformer: A unified architecture for predicting multiple agent trajectories. *arXiv preprint arXiv:2106.08417* (2021)
78. Nguyen, X.S.: Geomnet: A neural network based on riemannian geometries of spd matrix space and cholesky space for 3d skeleton-based interaction recognition. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 13379–13389 (2021)
79. Patrick, M., Campbell, D., Asano, Y., Misra, I., Metze, F., Feichtenhofer, C., Vedaldi, A., Henriques, J.F.: Keeping your eye on the ball: Trajectory attention in video transformers. *Advances in Neural Information Processing Systems* **34** (2021)
80. Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3d human pose estimation in video with temporal convolutions and semi-supervised training. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 7753–7762 (2019)
81. Perez, M., Liu, J., Kot, A.C.: Interaction relational network for mutual action recognition. *IEEE Transactions on Multimedia* (2021)
82. Perez, M., Liu, J., Kot, A.C.: Skeleton-based relational reasoning for group activity analysis. *Pattern Recognition* p. 108360 (2021)
83. Pramono, R.R.A., Chen, Y.T., Fang, W.H.: Empowering relational network by self-attention augmented conditional random fields for group activity recognition. In: *European Conference on Computer Vision*. pp. 71–90. Springer (2020)
84. Qi, M., Qin, J., Li, A., Wang, Y., Luo, J., Van Gool, L.: stagnet: An attentive semantic rnn for group activity recognition. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. pp. 101–117 (2018)
85. Rai, N., Chen, H., Ji, J., Desai, R., Kozuka, K., Ishizaka, S., Adeli, E., Niebles, J.C.: Home action genome: Cooperative compositional action understanding. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 11184–11193 (2021)
86. Ramanathan, V., Huang, J., Abu-El-Haija, S., Gorban, A., Murphy, K., Fei-Fei, L.: Detecting events and key actors in multi-person videos. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 3043–3053 (2016)

87. Sendo, K., Ukita, N.: Heatmapping of people involved in group activities. In: 2019 16th International Conference on Machine Vision Applications (MVA). pp. 1–6. IEEE (2019)
88. Shao, D., Zhao, Y., Dai, B., Lin, D.: Finegym: A hierarchical video dataset for fine-grained action understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2616–2625 (2020)
89. Shi, C., Holtz, C., Mishne, G.: Online adversarial purification based on self-supervised learning. In: International Conference on Learning Representations (2020)
90. Shu, T., Todorovic, S., Zhu, S.C.: Cern: confidence-energy recurrent network for group activity recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5523–5531 (2017)
91. Shu, T., Todorovic, S., Zhu, S.C.: Cern: confidence-energy recurrent network for group activity recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5523–5531 (2017)
92. Shu, X., Tang, J., Qi, G., Liu, W., Yang, J.: Hierarchical long short-term concurrent memory for human interaction recognition. IEEE transactions on pattern analysis and machine intelligence (2019)
93. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
94. Singh, K.K., Mahajan, D., Grauman, K., Lee, Y.J., Feiszli, M., Ghadiyaram, D.: Don't judge an object by its context: Learning to overcome contextual bias. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11070–11078 (2020)
95. Snower, M., Kadav, A., Lai, F., Graf, H.P.: 15 keypoints is all you need. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6738–6748 (2020)
96. Socher, R., Karpathy, A., Le, Q.V., Manning, C.D., Ng, A.Y.: Grounded compositional semantics for finding and describing images with sentences. Transactions of the Association for Computational Linguistics **2**, 207–218 (2014)
97. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014)
98. Sun, J.J., Zhao, J., Chen, L.C., Schroff, F., Adam, H., Liu, T.: View-invariant probabilistic embedding for human pose. In: European Conference on Computer Vision. pp. 53–70. Springer (2020)
99. Sun, P., Wu, B., Li, X., Li, W., Duan, L., Gan, C.: Counterfactual debiasing inference for compositional action recognition. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 3220–3228 (2021)
100. Surís, D., Liu, R., Vondrick, C.: Learning the predictability of the future. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12607–12617 (2021)
101. Tancik, M., Srinivasan, P.P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J.T., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. arXiv preprint arXiv:2006.10739 (2020)
102. Thilakarathne, H., Nibali, A., He, Z., Morgan, S.: Pose is all you need: The pose only group activity recognition system (pogars). arXiv preprint arXiv:2108.04186 (2021)

103. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
104. Vendrov, I., Kiros, R., Fidler, S., Urtasun, R.: Order-embeddings of images and language. arXiv preprint arXiv:1511.06361 (2015)
105. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al.: Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence (2020)
106. Wang, M., Ni, B., Yang, X.: Recurrent modeling of interaction context for collective activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3048–3056 (2017)
107. Wu, J., Wang, L., Wang, L., Guo, J., Wu, G.: Learning actor relation graphs for group activity recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9964–9974 (2019)
108. Wu, L.F., Wang, Q., Jian, M., Qiao, Y., Zhao, B.X.: A comprehensive review of group activity recognition in videos. International Journal of Automation and Computing pp. 1–17 (2021)
109. Xu, D., Fu, H., Wu, L., Jian, M., Wang, D., Liu, X.: Group activity recognition by using effective multiple modality relation representation with temporal-spatial attention. IEEE Access **8**, 65689–65698 (2020)
110. Yan, R., Tang, J., Shu, X., Li, Z., Tian, Q.: Participation-contributed temporal dynamic model for group activity recognition. In: Proceedings of the 26th ACM international conference on Multimedia. pp. 1292–1300 (2018)
111. Yan, R., Xie, L., Shu, X., Tang, J.: Interactive fusion of multi-level features for compositional activity recognition. arXiv preprint arXiv:2012.05689 (2020)
112. Yan, R., Xie, L., Tang, J., Shu, X., Tian, Q.: Higcin: hierarchical graph-based cross inference network for group activity recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
113. Yan, R., Xie, L., Tang, J., Shu, X., Tian, Q.: Social adaptive module for weakly-supervised group activity recognition. In: European Conference on Computer Vision. pp. 208–224. Springer (2020)
114. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-second AAAI conference on artificial intelligence (2018)
115. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: Reppoints: Point set representation for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9657–9666 (2019)
116. Yuan, H., Ni, D.: Learning visual context for group activity recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 3261–3269 (2021)
117. Yuan, H., Ni, D., Wang, M.: Spatio-temporal dynamic inference network for group activity recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7476–7485 (2021)
118. Yun, K., Honorio, J., Chattopadhyay, D., Berg, T.L., Samaras, D.: Two-person interaction detection using body-pose features and multiple instance learning. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. pp. 28–35. IEEE (2012)
119. Zappardino, F., Uricchio, T., Seidenari, L., Del Bimbo, A.: Learning group activities from skeletons without individual action labels. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 10412–10417. IEEE (2021)

120. Zhan, Y., Yu, J., Yu, T., Tao, D.: Multi-task compositional network for visual relationship detection. *International Journal of Computer Vision* **128**(8), 2146–2165 (2020)
121. Zhao, L., Wang, Y., Zhao, J., Yuan, L., Sun, J.J., Schroff, F., Adam, H., Peng, X., Metaxas, D., Liu, T.: Learning view-disentangled human pose representation by contrastive cross-view mutual information maximization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12793–12802 (2021)
122. Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., Xie, X.: Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 30 (2016)

COMPOSER: Compositional Reasoning of Group Activity in Videos with Keypoint-Only Modality

Supplementary Material

Honglu Zhou¹, Asim Kadav², Aviv Shamsian³, Shijie Geng¹, Farley Lai², Long Zhao⁴, Ting Liu⁴, Mubbashir Kapadia¹, and Hans Peter Graf²

¹ Department of Computer Science, Rutgers University, Piscataway, NJ, USA

{hz289,sg1309,mk1353}@cs.rutgers.edu

² NEC Laboratories America, Inc., San Jose, CA, USA

{asim,farleylai,hpg}@nec-labs.com

³ Bar-Ilan University, Israel

aviv.shamsian@biu.ac.il

⁴ Google Research, Los Angeles, CA, USA

{longzh,liuti}@google.com

This appendix is organized as follows:

- A Results Using Different Num. of Prototypes
- B Ablation Study
- C Efficiency Comparison
- D Additional Qualitative Results
- E Confusion Matrices and Failure Cases
- F Method and Implementation Details
- G Extended Discussion on Related Work
- H Discussion of COMPOSER (e.g., Societal Impact)

A Results Using Different Num. of Prototypes

In Table 1, we evaluate the impact of the number of prototypes K (i.e., the number of clip clusters) that is used for contrastive clustering learning on the GAR accuracy of COMPOSER. We use the Original split of the Volleyball dataset for this evaluation. We observe that varying the number of prototypes does not affect much the performance. The performance first improves as the number of prototypes increases, then decreases as the number of prototypes keeps increasing. The number of prototypes has little influence as long as it is reasonably “enough”. The practice is to set the number of prototypes at least one order of magnitude larger than the true number of classes in the dataset [11]. Hence, for simplicity, we do not spend extensive efforts in fine-tuning COMPOSER w.r.t. this

Table 1. Impact of the number of prototypes. GAR accuracy of COMPOSER on the Original split of the Volleyball dataset using different number of prototypes.

Number of Prototypes	10	50	100	1,000	5,000	10,000
GAR Accuracy (%)	94.02	94.54	94.69	94.62	94.54	94.32

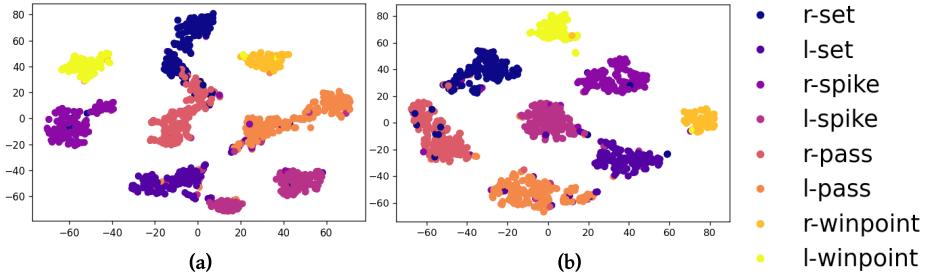


Fig. 1. t-SNE visualizations on the Volleyball dataset show that the clip embedding space learned by COMPOSER using different number of prototypes: (a) 10 prototypes, and (b) 1000 prototypes. Best viewed in color. More number of prototypes can lead to a better separation of the clips in distinct group activity classes.

hyper-parameter; results reported in the main paper are from COMPOSER trained using 1,000 prototypes for all datasets and splits.

We visualize the clip embedding space (after t-SNE 2D projection) learned by COMPOSER using 10 prototypes and 1000 prototypes in Fig. 1 (a) and (b), respectively. We take the representation of the CLS token from the last scale and the last Multiscale Transformer block as the representation of the clip to produce the embedding space visualization. In Fig. 1, each dot represents a test clip and the color of the dot indicates the group activity label of the clip. A higher number of prototypes can lead to a better grouping of clips with the same group activity class, as well as a better separation of clips in different group activity classes; this accords with the quantitative results shown in Table 1.

B Ablation Study

We conduct ablation experiments to verify the effectiveness of proposed techniques; results are in Table 2 (methodology details are in Appendix F.4).

Contrastive clustering and scale agreement regularize the multiscale representations. As demonstrated in Table 2, performance drops without the contrastive clustering learning. The swapped prediction setup helps the model to maintain consistency across representations of the multiple scales of the same clip, which regularizes the intermediate representations.

We also experiment with the ‘Label Consistency’ method [89] that minimizes the L_2 distance between 2 views of an instance in the logit space. Replacing

Table 2. Ablation study of COMPOSER on Volleyball original split. The ablation study verifies the effectiveness of each proposed technique.

Ablation	Test Acc. (%) ↑
No Clustering	93.4
Label Consistency for Scale Agreement	93.9
1-Scale: Keypoint	91.2
2-Scale: Keypoint + Person	93.2
3-Scale: Keypoint + Person + Interaction	93.9
No Actor Dropout	94.0
No Horizontal Flip	94.0
No Horizontal Move	94.2
No Vertical Move	94.2
No Auxiliary Prediction	93.3
No Multiscale Transformer	88.1
Transformer Encoder Reverse Order	86.8
Transformer Encoder Parameter Sharing	93.4
All Tokens to One Transformer Encoder	92.4
Time-Varying Person Grouping	92.8
COMPOSER (our full model)	94.6

contrastive clustering with Label Consistency for scale agreement, result is better than the previous ablation, but worse than **COMPOSER**. Better performance of **COMPOSER** can be attributed to the additional benefits of the clustering loss, which draws clips that are semantically related close together by comparing with the prototypes. Both experiments indicate that encouraging scale agreement can bring benefits for the multi-scale learning models, which is unfortunately neglected by prior works.

Greater number of scales yields more information and an effective scale agreement. We find that increasing the number of scales leads to a higher accuracy. More scales indicates more information about the entities in the scene. Besides, given more scales, hierarchical representations are able to be better maintained, and techniques such as contrastive clustering and auxiliary predictions are more effective.

Data augmentations increase the training data size and inject benign noises, leading to generalization. Results in Table 2 show the gains brought by each data augmentation technique described in Sec. 3.4 of the main paper. Among the four types of data augmentation, Horizontal Flip (which is commonly used by existing works [119,102]) and Actor Dropout are the most critical ones. Even though data augmentation is less effective than other techniques proposed, it increases the training data size and injects noises that help model to generalize.

Auxiliary prediction aids learning the intermediate representations. For the ‘No Auxiliary Prediction’ ablation, the loss of group activity is only computed from the clip token from the last scale of the last Multiscale Transformer (note that the person action loss and clustering loss are still in use).

Performance of this ablation largely drops, indicating auxiliary prediction is a simple but yet effective technique.

Multiscale Transformer learns higher-level knowledge of the video by compositionally reasoning over concepts from finest-grained to coarsest-grained. We design an ablation to remove Multiscale Transformer. Specifically, the group activity classifier directly takes features of the object token and person tokens (learned from features of keypoint tokens) as an input, and the person action classifier takes the features of person tokens as an input. The outcome of this ablation is worse performance than the 1-scale ablation that does not even use person features and person action labels – which indicates the importance of relational reasoning.

Misc. Unlike previous works [27,58], person-to-group association mechanism is not our focus. Hence, we use heuristics [82] for the Volleyball dataset, K-means [65] for the Collective Activity dataset, and set $g!=2$ without tuning. With K-means, actors are mapped to groups adaptively in each Multiscale Transformer block as person representations are refined. We have experimented with *time-varying* person grouping with K-means, and the result is **92.8%** on the original split of the Volleyball dataset. For this experiment, since we need person and group representations at each timestamp, the number of tokens is increased (multiplied by T) for the 2 transformer encoders, which potentially leads to issues such as over-smoothing [117] and raises the challenge for attention. We hypothesize that a carefully-crafted mechanism for learning spatio-temporal relations is required for time-varying person grouping.

We have experimented with Multiscale Transformer variations. In Multiscale Transformer, 4 different transformer encoders separately model the contextual information of each scale, which eases learning (because the information granularity and features across scales can vary significantly). Parameter-sharing across the 4 encoders yielded a result of **93.4%** on the original split of the Volleyball dataset, and feeding all tokens to one encoder obtained **92.4%**. The order of encoders in Multiscale Transformer is in accordance with the hierarchy of person-related tokens, from fine to coarse. This allows COMPOSER to *compose* the high-level representations from low-level ones and distill the knowledge. One can perform grid search to find the optimal order of the 4 encoders, but that will lead to 24 permutations which require lots of resources and runtime. We have performed an experiment with the reverse order (representations of coarser tokens are broadcast to finer tokens), and the result is **86.8%**, which is much worse.

C Efficiency Comparison

Backbone efficiency comparison to support the keypoint-only setup. Backbones used by prior works vary *a lot* (see Table 4 and Table 5). Because AT [29] is a Transformer-based model like ours and it has reported result of using the keypoint-only modality, we follow AT to use the HRNet [105] as the person keypoint estimation backbone.

Table 3. Efficiency Comparison (FLOPs). Please see details in Appendix C.

Backbone						GAR Model	
HRNet	VGG-19	VGG-16	Inception-v3	ResNet-18	AlexNet	COMPOSER	GroupFormer [65]
0.9 T	3.6 T	2.8 T	0.7 T	0.3 T	0.1 T	297 M	595 M

*Note: ‘T’ stands for trillion and ‘M’ for million. Ours are marked in bold.

We also report the results of COMPOSER using POGARS’s person keypoint estimation backbone – Hourglass [?] since POGARS [102] is the keypoint-only method that has the closest result to ours on the Volleyball dataset. Comparing COMPOSER with POGARS (both use Hourglass as the backbone): on VD Olympic split COMPOSER 92.9% v.s. POGARS 89.7%; and on VD Original split COMPOSER 94.3% v.s. POGARS 93.9%. The superiority of COMPOSER is not affected because COMPOSER is able to address noisy estimated keypoints and disregard inaccurate keypoints by modeling attention over the keypoints (unlike POGARS or AT that just model attention at the person scale).

To support the keypoint-only setup, we compute the FLOPs for HRNet (our keypoint backbone) and RGB backbones used by prior works when obtaining features of all persons in a clip on the Volleyball dataset. FLOPs are: HRNet **0.9T**, VGG-19 [93] **3.6T**, VGG-16 [93] **2.8T**, Inception-v3 [?] **0.7T**, ResNet-18 [37] **0.3T**, and AlexNet [?] **0.1T** (Table 3). VGG-19 and VGG-16 are more computational expensive than HRNet. Our method is agnostic to the type of the keypoint estimation backbone and robust w.r.t. noisy estimated keypoints. Therefore, real-time applications can use an efficient backbone.

Efficiency comparison with SOTA GAR methods. We also compare FLOPs of prior Transformer-based GAR methods with ours. For a fair comparison, the methods all have 2 blocks of their respective GAR reasoning module (e.g., Multiscale Transformer in COMPOSER and the CSTT module in GroupFormer) and share hyper-parameters of the Transformers inside (e.g., dimension of the FFN layer inside Transformer). In addition, computation spent on person feature extraction is excluded since different backbones can be used.

FLOPs are (given a Collective Activity dataset’s input): Ours **297M**, GroupFormer [65] **595M**, and AT [29] **17M** where ‘M’ stands for million. While AT is the most efficient one, its efficacy (only 92.8% on Collective Activity) and generalization are unsatisfactory. GroupFormer is the most computational expensive one due to 5 Transformers inside its CSTT module and leveraging image scenes – even we only report FLOPs for its most basic version that leverages the least signals possible (RGB + Scene).

Note that the COMPOSER variant that uses RGB modality (mentioned in the main paper) has **127M** FLOPs and is more efficient than COMPOSER that uses keypoints because the former only models 3 scales (person, interaction and group). To further reduce latency for real-life applications (e.g., on-device scenarios), we can have a light-weight COMPOSER variant that only models one scale during inference or uses smaller hidden dimensions and yet retains the most efficacy and generalization by using techniques like knowledge distillation [?].

Table 4. Detailed comparisons between our results and the reported SOTA methods’ results on the **Original** split of the **Volleyball** dataset. We have made an effort to collect and list results of prior works using various backbones and modalities. Note that the backbones used by prior works vary a lot. The top 3 performance scores are highlighted as: **First**, **Second***, **Third**. COMPOSER outperforms the latest GAR methods that use a single modality (+0.7% improvement), and performs favorably compared against methods that exploit multiple expensive modalities (Appendix C)

Method	Modality				Backbone		Acc. ↑ (%)
	Keypoint	RGB	Flow	Scene	Keypoint	RGB/Flow/Scene	
HDTM [47]		✓			–	AlexNet	81.9
CERN [90]		✓			–	VGG-16	83.3
stagNet [84]		✓			–	VGG-16	89.3
RCRG [45]		✓			–	VGG-19	89.5
SSU [8]		✓			–	Inception-v3	90.6
PRL [41]		✓			–	VGG-16	91.4
AT [29]		✓			–	I3D	91.4
ARG [107]		✓			–	VGG-16	91.9
		✓			–	Inception-v3	92.5
		✓			–	VGG-19	92.6
HiGCIN [112]		✓			–	AlexNet	88.6
		✓			–	ResNet-18	91.5
DIN [117]		✓			–	ResNet-18	93.1
		✓			–	VGG-16	93.6
Zappardino <i>et al.</i> [119]	✓				OpenPose	–	91.0
GIRN [82]	✓				OpenPose	–	92.2
AT [29]	✓				HRNet	–	92.3
POGARS [102]	✓				Hourglass	–	93.9
CRM [5]		✓	✓		–	I3D	93.0
Ehsanpour <i>et al.</i> [27]		✓		✓	–	I3D	93.1
AT [29]	✓	✓	✓		–	I3D	93.0
	✓	✓	✓		HRNet	I3D	93.5
	✓	✓	✓		HRNet	I3D	94.4
GIRN [82]	✓	✓			OpenPose	Inception-v3	93.5
	✓	✓			OpenPose	Inception-v3	93.0
	✓	✓			OpenPose	Inception-v3	94.0
TCE+STBiP [116]	✓			✓	HRNet	Inception-v3	92.9
	✓		✓		–	Inception-v3	93.3
	✓		✓		HRNet	Inception-v3	94.1
	✓		✓		HRNet	VGG-16	92.9
	✓		✓		–	VGG-16	94.1
SACRF [83]	✓	✓	✓	✓	HRNet	VGG-16	94.7
	✓	✓	✓	✓	–	I3D	94.5
	✓	✓	✓	✓	AlphaPose	I3D	95.0*
GroupFormer [65]		✓	✓	✓	–	Inception-v3	94.1
		✓	✓	✓	–	I3D	94.9
		✓	✓	✓	AlphaPose	I3D	95.7
COMPOSER (ours)		✓			Hourglass	–	94.3
		✓			HRNet	–	94.6

*Note: “Flow” denotes optical flow input, and “Scene” denotes additional image context features of the entire frames (*fewer checks are better*). Yellow shaded rows highlight that the methods use just the RGB-based input, whereas blue for just keypoint.

Table 5. Detailed comparisons between our results and the reported SOTA methods’ results on the **Collective Activity** dataset. The top 3 performance scores are highlighted as: *First*, *Second**, *Third*. COMPOSER outperforms the latest GAR methods that use a single modality (+2.8% improvement), and performs favorably compared against methods that exploit multiple expensive modalities (ours is the second best)

Method	Modality				Backbone		Acc. ↑ (%)
	Keypoint	RGB	Flow	Scene	Keypoint	RGB/Flow/Scene	
HDTM [47]		✓			–	AlexNet	81.5
CERN [90]		✓			–	VGG-16	87.2
stagNet [84]		✓			–	VGG-16	89.1
ARG [107]		✓			–	VGG-16 Inception-v3	90.1 91.0
HiGCIN [112]		✓			–	AlexNet ResNet-18	92.5 93.4
CRM [5]		✓	✓		–	I3D	85.8
Ehsanpour <i>et al.</i> [27]		✓		✓	–	I3D	89.4
AT [29]	✓	✓			HRNet	I3D	91.0
	✓		✓		HRNet	I3D	91.2
	✓	✓			–	I3D	92.8
SACRF [83]	✓	✓	✓	✓	–	I3D	94.6
GroupFormer [65]	✓	✓	✓	✓	AlphaPose	I3D	95.2
	✓	✓	✓	✓	–	Inception-v3	93.6
	✓	✓	✓	✓	–	I3D	94.7
COMPOSER (ours)				✓	HRNet	–	96.2*

*Note: “Flow” denotes optical flow input, and “Scene” denotes additional image context features of the entire frames (*fewer checks are better*). Yellow shaded rows highlight that the methods use just the RGB-based input, whereas blue for just keypoint. We are the first to report the result of a keypoint-only method on this dataset.

D Additional Qualitative Results

We visualize the attention matrices produced by the last Multiscale Transformer block in COMPOSER of CAD test clips in different group activity classes in Fig. 2, 5, 6 and 7. In Fig. 5 of the main paper, we visualize the attention matrices of a VD test clip in group activity class “pass”. We provide the same visualization for the other 3 main group activity classes of VD in Fig. 8, 9 and 10.

In the figures, we highlight the tokens that COMPOSER has mostly attended to at each scale (darker color denotes larger attention weights). Each figure contains rich information. Please zoom in on the images to appreciate the details. Here, we summarize the main findings:

1. Actor-related tokens associated with the key person(s) are often identified as the most important tokens for the CLS and object token by COMPOSER

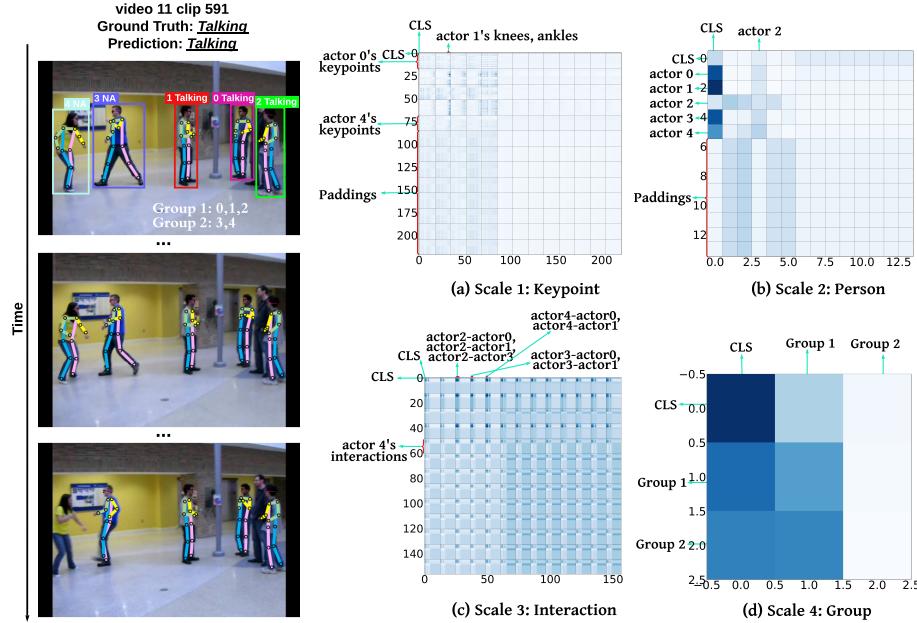


Fig. 2. Qualitative results of COMPOSER on CAD – showcasing attention matrices of a test set instance in the “talking” class. The person to person group mapping identified by COMPOSER, estimated keypoints from the backbone, and the ground truth person bounding boxes with action labels are also shown on the left side of the figure.

(e.g., Fig. 8, 9, etc). In certain cases, the object token is also identified as important (e.g., as shown in Fig. 9, the object token ‘ball’ is identified as important at scale 1, 3 and 4). COMPOSER is able to attend to relevant entities across different scales and can produce interpretable results.

2. COMPOSER learns to recognize the group activities based upon the unique characteristics of each group activity class. E.g., in Fig. 10, the pattern of the attention weights of the *winpoint* class at scale 3 is quite different from the pattern observed in the other 3 main group activity classes in VD. Recognition of the other group activities in VD (e.g., *spike*), is often determined by a key player who is performing the key action (e.g., *spiking*). In contrast, the *winpoint* group activity is not heavily dependent on one or two key players; instead, it is defined by the overall person-to-person interactions of the team that just scored. As shown in Fig. 10, person-to-person interaction tokens formed by players in the right team tend to closely attend to each other.
3. On CAD, COMPOSER has identified interesting person-to-group mappings as shown in Fig. 2, 5, etc. On VD, we find that at the group scale, the CLS and object tokens often mostly attend to each other, and the two teams often mostly attend to each other. We posit that this is because, at the previous three scales, the CLS and object tokens mostly attend to actor-related tokens.

The object token, which contains less correlated information, then becomes the most important signal at scale 4 for the CLS token in order to correctly recognize the group activity.

E Failure Cases and Confusion Matrices

E.1 Failure Case Analysis

Volleyball. Through examining failure cases of **COMPOSER**, we find numerous mislabeled test clips. We coin these cases as *false failure cases* (i.e., the annotated label is wrong but the prediction from **COMPOSER** is correct). Sometimes the annotation of the team is wrong (e.g., a test clip with the *left set* group activity is annotated as *right set* as shown in Fig. 11), and sometimes the annotated main group activity is wrong (e.g., a test clip with the *right set* group activity is annotated as *right spike* as shown in Fig. 12). As a result, the models that have achieved high accuracy on the Original split of VD might capture misleading noises and cannot generalize, and therefore the pursuit of beating SOTA might be pointless. For future research, proposing methods that make use of RGB signals but overcome scene biases on the Olympic split is a recommended direction.

We visualize two *true* failure cases of **COMPOSER** in Fig. 13 and Fig. 14. The wrong prediction of **COMPOSER** made on case shown in Fig. 13 could be attributed to the fact that the arms of actor 0 is occluded. In Fig. 14, **COMPOSER** fails to identify which team is performing the group activity *spike*. However, we notice a major change in the camera location of this test clip, which causes this test clip possibly to be difficult even for humans to make a correct prediction.

Collective Activity. We visualize several failure cases of **COMPOSER** on the test set of CAD in Fig. 15, 16, 17, and 18. Failures of **COMPOSER** can be attributed to severe occlusions (Fig. 15), misleading movement of actors (Fig. 16 and 17) and the current use of a relatively short temporal window for each clip (Fig. 18), which suggest ways to further improve **COMPOSER**, e.g., addressing the issue of severe occlusion, using more frames per clip with attention-based relational reasoning in both spatial and temporal domains, etc.

E.2 Confusion Matrix Analysis

We show the confusion matrices of **COMPOSER** in Fig. 3. On the Volleyball dataset (the Original split), for each group activity class **COMPOSER** achieves an accuracy over 90% with the lowest accuracy for the *right set* class. Most failures emerge from discriminating the *set*, *pass*, and *spike* activities which can be a result of highly similar actions or positions of the key player in some clips [29,41,112]. Occasionally, the model struggles to distinguish which team (left or right) performs the activity. We hypothesize that adding more object tokens/keypoints such as the keypoints of the net to **COMPOSER** may help to address this problem. Nevertheless, different camera positions of some clips in the dataset (e.g., Fig. 14) might cause the difficulty for a model to learn which team performs the activity.

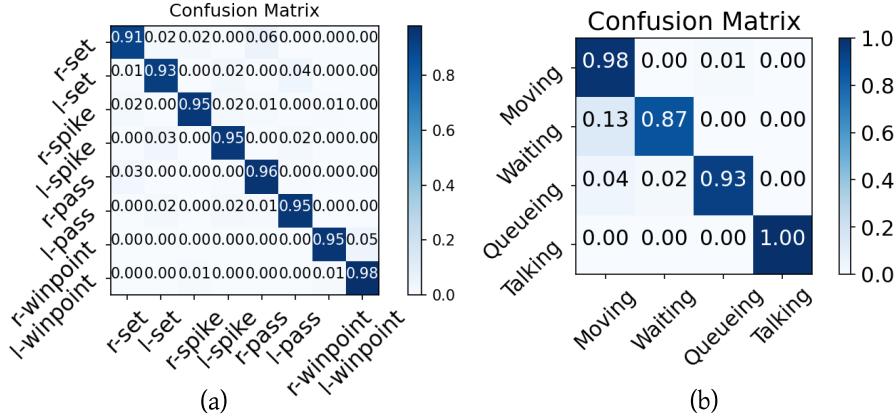


Fig. 3. Confusion matrices of COMPOSER on (a) the Volleyball dataset (the Original split) and (b) the Collective Activity dataset.

On the Collective Activity dataset, COMPOSER occasionally mistakes *waiting* to *moving*, which may because the current temporal dynamics of clips is too short to catch differences between the two classes [116] and a lot more examples of *moving* than *waiting* in the dataset (more than twice more).

F Method and Implementation Details

F.1 Transformer

We briefly describe the Transformer encoder [103] used in Multiscale Transformer in this subsection. The basic components of the Transformer encoder include 1) Multi-head Self-Attention (MSA), 2) Multi-Layer Perceptron (MLP), and 3) Skip Connection [37], Dropout [97] and Layer Normalization [7] (Add & Dropout & LN).

MSA. Central to the Transformer encoder is the self-attention function. In the self-attention function, the input $X \in \mathbb{R}^{n \times d}$ is first linearly transformed to three parts, i.e., *query* $Q \in \mathbb{R}^{n \times d_k}$, *key* $K \in \mathbb{R}^{n \times d_k}$ and *value* $V \in \mathbb{R}^{n \times d_v}$, where n denotes the number of tokens in the input sequence, and d , d_k and d_v are the representation dimensions of the input, query (or key), and value, respectively. The Scaled Dot-Product Attention is applied on Q , K , and V :

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \quad (1)$$

Then, a linear layer is used to produce the output. We use h paralleled heads of the Scaled Dot-Product Attention to increase the representation power. Specifically, MSA splits the query, key and value for h times and performs the self-attention function in parallel, and then the output values of each head are concatenated and linearly projected to form the final output [35].

MLP. The MLP is for feature transformation and non-linearity:

$$\text{MLP}(X) = \sigma(XW_1 + b_1)W_2 + b_2 \quad (2)$$

where $W_1 \in \mathbb{R}^{d \times d_m}$ and $W_2 \in \mathbb{R}^{d_m \times d}$ are weights of the two fully-connected layers, $b_1 \in \mathbb{R}^{d_m}$ and $b_2 \in \mathbb{R}^d$ are the bias terms, and $\sigma(\dots)$ is the non-linear activation function such as ReLU [30] or GELU [38].

Add & Dropout & LN. The output from MSA (or MLP) is added with the input of MSA (or MLP) to enforce the skip connection. Then, a dropout layer is used, followed by the Layer Normalization (LN) that enables stable training and faster convergence. Layer normalization is applied over each sample $x \in \mathbb{R}^d$ as follows:

$$LN(x) = \frac{x - \mu}{\delta} \circ \gamma + \beta \quad (3)$$

where $\mu \in \mathbb{R}$ and $\delta \in \mathbb{R}$ are the mean and standard deviation of the features, respectively, \circ denotes the element-wise multiplication, $\gamma \in \mathbb{R}^d$ and $\beta \in \mathbb{R}^d$ are learnable affine transform parameters for scaling and shifting, respectively.

Given the input X , the computations in order in the Transformer encoder are: MSA, Add & Dropout & LN, MLP, and Add & Dropout & LN.

F.2 Keypoint Initial Representation

Keypoint is the finest-grained actor-related entity that we consider for GAR. In this subsection, we describe the keypoint initial representation (i.e., representations of the input keypoint tokens of the first Multiscale Transformer layer). As described in the main paper, representations of actor-related tokens in coarser scales are learned and aggregated from that of the finer scales.

A keypoint in a frame has the information of keypoint type, as well as keypoint coordinate in both of the image space and the time space. We use three GCN [56] layers to map each person keypoint type into a learned vector in order to encode the intrinsic connections of different keypoint types. We apply feature standardization to the raw 2D keypoint coordinate and the temporal difference of coordinates in consecutive two frames. We also normalize the keypoint coordinate in a person-wise manner to account for rotation, translation, and scale differences [119,98]. In addition, we use the Learned Fourier Positional Encoding [101] to map each image coordinate into a learned vector, and use the Learned Absolute Positional Encoding to learn a vector for each time coordinate. To mitigate the issue of noisy estimated keypoints, we use the temporal *Object Keypoint Similarity* (OKS) proposed in [95], and use the mean OKS scores of each person as additional features. The above procedure is summarized in Fig. 4. These features are concatenated to form the initial composite representation of a person keypoint in a frame, and a keypoint token is represented by concatenation and Feed Forward Network (FFN) based transformation of keypoints’ representations in all timestamps.

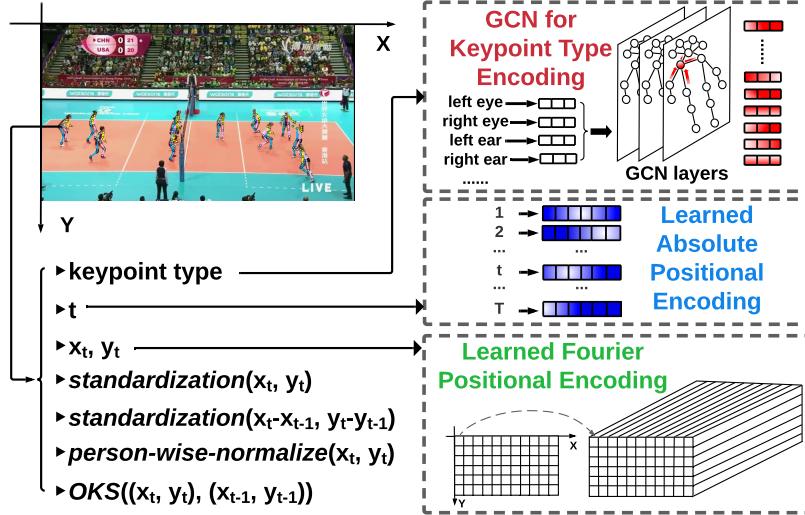


Fig. 4. Representation of a person keypoint in a particular frame.

F.3 Implementation Details of COMPOSER

For annotation parsing and video preprocessing (e.g., obtaining clips from a long CAD video), we follow prior works [117,107]. On the Volleyball dataset, annotations of group activity, players’ bounding boxes and their actions in the middle frame of each clip are provided. Person tracklet data is provided by [87]. For the Collective Activity dataset, annotations include actors’ bounding boxes and their action labels on the center frame of every ten frames and group activity labels for every ten frames. In order to make a fair comparison with related works [116,117,29,107], we use $T = 10$ frames as the input to our model for both training and testing on both datasets.

We use HRNet [105]⁵ to obtain estimated person keypoints following [29,116] ($j' = 17$). There are a total of 17 different keypoint types: nose, left eye, right eye, left ear, right ear, left shoulder, right shoulder, left elbow, right elbow, left wrist, right wrist, left hip, right hip, left knee, right knee, left ankle and right ankle. On the Volleyball dataset, the maximal number of actors in a video $p' = 12$, and on Collective Activity, $p' = 13$. On both datasets, the number of person groups per video $g' = 2$. On Volleyball, actors are grouped into the two person groups by heuristics, i.e., according to the horizontal positions of the actors, the left most 6 actors form a sub-group and the rest actors form the other sub-group. We find that using clustering algorithms such as K-means (given the coordinates of the actors as features) can generate similar results as the heuristics on the Volleyball dataset. Hence, we choose to use the heuristics for simplicity for Volleyball, and

⁵ <https://github.com/leoxiaobin/deep-high-resolution-net.pytorch>

use K-means⁶ to form person groups on the Collective Activity dataset (given the input of the learned person representations from **COMPOSER**).

On the Volleyball dataset, the object keypoints are from the ball trajectories annotated by [82] ($e' = 1$)⁷. The initial representation of the object keypoint is similar to that of the person keypoint, i.e., a concatenation of the time positional encoding, Fourier positional encoding, and standardized keypoint coordinate and temporal difference. The initial object token is formed by concatenation and FFN-based transformation of object keypoints’ representations in all timestamps. On the other hand, we do not use the object token for the Collective Activity dataset because the Collective Activity dataset does not have any human-object interactions.

On both datasets, the number of Multiscale Transformer blocks M is set as 2, the number of attention heads of the Transformer encoder at each scale is set to 2, 8, 2 and 2, respectively, and the dropout rate of the Transformer encoder at each scale is set to 0.5, 0.2, 0.2 and 0, respectively. We find that a smaller dropout rate in the coarser scales tends to yield a better performance. The dimension of the MLP layer in all Transformer encoders is set to 1024 (i.e., $d_m = 1024$), and the non-linear activation function is ReLU. The hidden dimension $d = 256$ (d_k , d_v and d are equal) on Volleyball and $d = 128$ on Collective Activity. Because we focus on semantic relational reasoning over temporal relation capturing, we use MLPs with 1 hidden layer of dimensionality d to flatten out the time axis for each entity – in this way, track-based representations are formed for each entity. To aggregate the token representations from a finer scale to the coarser scale, FFNs are used when the number of tokens for aggregation at that scale is fixed (e.g., 2 persons aggregates to an interaction), otherwise summation is used (e.g., the number of persons to form a group varies on the Collective Activity dataset due to K-means). The cross entropy loss is used during training for both group activity and person action classification. We use the Adam optimizer [55] and train the model for 45 epochs with an initial learning rate 0.0005 and decrease the learning rate to 0.0001 at epoch 40. The weight decay is 0.001, λ is 3, and batch size is 256. Following [11], the temperature parameter $\tau = 0.1$, $\varepsilon = 0.05$, and the number of iterations of the Sinkhorn-Knopp algorithm⁸ is set to 3. The number of prototypes K is 1000 for all experiments except the ones described in Appendix A. In data augmentations, the range of random perturbation is set to 1 pixel location. We use the PyTorch⁹ Python library.

F.4 Implementation Details of Ablation Studies

In this subsection, we describe the methodology of the ablations that we present in Appendix B in details. For all of the ablations, we use the same set of hyper-parameters as our full model.

⁶ https://github.com/subhadarship/kmeans_pytorch

⁷ For real-world data, one can resort to ball trajectory extraction [1,79] for sports videos or object keypoint detection tools [42,9,69].

⁸ <https://github.com/facebookresearch/swav>

⁹ <https://pytorch.org>

No Clustering. The only difference from this ablation to our full model **COMPOSER** is the loss function used in training. Instead of using Equation (6) (in the main paper) as the loss function, this ablation uses the following loss function:

$$\mathcal{L}_{\text{total}} = \sum_{m=1}^{M-1} \mathcal{L}_{\text{groupAux}} + \lambda (\mathcal{L}_{\text{groupLast}} + \mathcal{L}_{\text{person}}) \quad (4)$$

Label Consistency [89] for Scale Agreement. Similar to the previous ablation, the only difference from this ablation to our full model **COMPOSER** is the loss function used in training, which is formulated as follows:

$$\mathcal{L}_{\text{total}} = \sum_{m=1}^{M-1} \mathcal{L}_{\text{groupAux}} + \lambda (\mathcal{L}_{\text{groupLast}} + \mathcal{L}_{\text{person}} + \mathcal{L}_{\text{consistency}}) \quad (5)$$

where $\mathcal{L}_{\text{consistency}}$ represents the loss term that minimizes the L_2 distance between 2 scales of a clip in the logit space. Specifically, for every two pairs of scales, we compute the L_2 loss given the two sets of GAR logits of the two scales. The $\mathcal{L}_{\text{consistency}}$ term is the mean of such L_2 losses over all pairs of scales.

1-Scale: Keypoint. For this ablation, there is only one Transformer encoder in the Multiscale Transformer block, and the tokens to the Transformer encoder are:

$$\text{Scale 1: } \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{k}_1^1, \dots, \mathbf{k}_{p'}^{j'}\}. \quad (6)$$

Since only the keypoint tokens are refined by the Multiscale Transformer block in this ablation and there is only 1 scale, this ablation uses the following loss function:

$$\mathcal{L}_{\text{total}} = \sum_{m=1}^{M-1} \mathcal{L}_{\text{groupAux}} + \lambda \mathcal{L}_{\text{groupLast}} \quad (7)$$

2-Scale: Keypoint + Person. For this ablation, there are two hierarchical scales in the Multiscale Transformer block, and the tokens to the Transformer encoders are:

$$\begin{aligned} \text{Scale 1: } & \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{k}_1^1, \dots, \mathbf{k}_{p'}^{j'}\}, \\ \text{Scale 2: } & \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{p}_1, \dots, \mathbf{p}_{p'}\}. \end{aligned} \quad (8)$$

This ablation uses the same loss function as our full model except that the number of pairs of scales for swapped prediction is only 1, i.e., pair scale1-scale2.

3-Scale: Keypoint + Person + Interaction. For this ablation, there are three hierarchical scales in the Multiscale Transformer block, and the tokens to the Transformer encoders are:

$$\begin{aligned} \text{Scale 1: } & \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{k}_1^1, \dots, \mathbf{k}_{p'}^{j'}\}, \\ \text{Scale 2: } & \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{p}_1, \dots, \mathbf{p}_{p'}\}, \\ \text{Scale 3: } & \{[\text{CLS}], \mathbf{e}_1, \dots, \mathbf{e}_{e'}, \mathbf{i}_1, \dots, \mathbf{i}_{p' \times (p'-1)}\}. \end{aligned} \quad (9)$$

This ablation uses the same loss function as our full model except that the number of pairs of scales for swapped prediction is 3, i.e., pair scale1-scale2, pair scale1-scale3, and pair scale2-scale3.

No Auxiliary Prediction. The only difference from this ablation to our full model **COMPOSER** is the loss function used in training. This ablation uses the following loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{groupLast}} + \mathcal{L}_{\text{person}} + \mathcal{L}_{\text{cluster}} \quad (10)$$

No Multiscale Transformer. In this ablation, the group activity classifier simply takes features of the *initial* object token and person tokens as inputs, and the person tokens are aggregated from the *initial* representations of keypoint tokens through concatenation and FFN. In addition, features of the person tokens are the inputs to the person action classifier. No Transformers are used in this ablation. Due to the lack of relational reasoning performed at multiple scales, this ablation uses the following loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{groupLast}} + \mathcal{L}_{\text{person}} \quad (11)$$

F.5 Miscellaneous

The Original split of the Volleyball dataset allows the GAR method to leverage scene biases in order to achieve a high accuracy, on the other hand, the Olympic split can better test the model generalization ability. None of the existing RGB-based GAR methods have performed experiments using the Olympic split. To obtain the results of prior RGB-based GAR methods on the Olympic split of the Volleyball dataset (Table 1 in the main paper), our implementations of PCTDM [110], SACRF [83], AT [29], ARG [107], TCE-STBiP [116] and DIN [117] are based on the public available codebase¹⁰¹¹ and we have verified the implementations through obtaining the results of these methods on the Original split of the Volleyball dataset and then comparing with the reported results from authors of each method. For VGG-16 [93], we use RoIAlign [36]¹² to obtain the person regional features and then use these person features to predict the individual actions and the group activity. All of these RGB-based methods (including our **COMPOSER** RGB-based variant) use the RGB-only modality and share the same VGG-16 backbone.

G Extended Discussion on Related Work

Group Activity Recognition. Early work on GAR relies on handcrafted features [20,61,34,16,21,18,74]; yet notable progress has been made by Deep Learning (DL) based approaches [47,25]. We review DL-based methods and refer readers to the comprehensive review of GAR presented in [108].

¹⁰ <https://github.com/JacobYuan7/DIN-Group-Activity-Recognition-Benchmark>

¹¹ <https://github.com/wjchaoGit/Group-Activity-Recognition>

¹² <https://github.com/longcw/RoIAlign.pytorch>

Early DL-based methods use Convolutional Neural Networks (CNNs) to extract the low-level visual features and then apply Recurrent Neural Networks such as LSTM [40] for temporal modeling [106,91,66,53]. Since learning inter-person interactions is essential for GAR [108], much of the recent research explores how to capture the contextual information about the actor and their relations [46,6,107,41,83]. Several works tackle this problem from a graph-based perspective [46,71,113,112] such as applying Graph Convolutional Networks (GCNs) [56] for deep relationship modeling [107]. More recent works utilize attention modeling [84,109,71,116] including using Transformers [29,65] to perform relational reasoning, with a focus on determining the most critical persons [107,29,83,116], groups [27,65], or interactions [113]. Existing works in the field of GAR have primarily used RGB- and/or optical-flow-based features with RoIAlign [36] to represent actors [112,84,107,8]. A few recent works replace or augment these features with keypoints/poses of the actors [70,14,102,65]. Some only use the numerical coordinate-based keypoint representation [119,102,82,83] while others use a high-dimensional vector from a deep pose backbone [29,116] which is not as efficient. In this paper, we use Transformers [103] for higher-order relationship modeling and use only the light-weight coordinate-based keypoint representation. Our work differs from prior methods in that we propose a Multiscale Transformer block to hierarchically reason about entities at different semantic scales and we aid learning group activities by improving the multiscale representations.

Action Recognition and Keypoint-based Prediction. Action Recognition is one of the primary tasks in video understanding. There has been rapid progress in recent years, starting from recognition of the low-level atomic actions performed by an individual (e.g., hand-waving, dancing, jumping), to paired-actions being acted by two persons [81,92,118,57] (e.g., shaking hands, hugging, punching), towards group activities that encompass many actors at once [20,47,86,82] (e.g., attack and defense in a sports game, pedestrians queuing). Our paper focuses on the most spatially complex scenarios, where multiple interacting individuals form the group activity. In addition, keypoint-based action recognition has drawn much attention [64,114,122,78,121]. Keypoint-based representation can be regarded as a high-level representation for dynamic behaviors, and is preferred due to benefits such as being compact and robust to variations of viewpoints, appearances, and surrounding distractions [23,67]. We study keypoint-based group activity recognition. We propose to use techniques including auxiliary prediction and data augmentations that can aid learning group activity from the keypoint modality.

Compositionality and Multiscale Learning. Compositionality is an active field of research in computer vision (CV) [120,54,111,99], natural language processing [96,104,48,24] and machine reasoning [44,43,10]. In terms of understanding videos centered on human actions, compositionality can be studied from different lenses, e.g., through formulating an activity as compositions of atomic actions temporally [85,32,15,72] or semantically [88,100], or decomposing actions by action-based aspects (verbs) and object components (noun) [50,51,73,62,72].

We tackle compositional video understanding by formulating a visual-semantic hierarchy, where each semantic hierarchy is regarded as representation of the video at a particular scale. Such an idea of multiscale learning has been a long-standing topic in CV as well [64,63,31,33]. Recently, researchers have started to introduce the concept of multiscale learning to Transformers [28,68,35] by operating self-attention over various scales of resolutions and/or channels, in order to obtain a multiscale pyramid of features often observed in CNNs. Distinct from prior works, we design **COMPOSER** that models semantic scene entities at different hierarchical scales to learn group activities effectively. **COMPOSER** is the first Transformer-based method with *explicit* multiscale modeling for GAR that improves the musicale representations with a contrastive clustering based objective.

H Discussion of COMPOSER

Motivation Intuitively, each scale provides enough information for GAR; only the information granularity varies, which causes recognition confidence to vary scale by scale. Therefore, we consider scales as different but correlated views of the clip, and utilize multiscale contrastive clustering learning (MCCL) to allow one scale to complement another. This allows learning better compositional structures and higher-order representations.

Pull close: Equation 2 in the main paper (swapped prediction) trains the model to produce multiscale representations of 1 clip such that the cluster assignment of the clip representation at one scale can be predicted from the clip representation at another scale, allowing representations of the same clip to be pulled close.

Pull away: If 2 clips are semantically different, as Equation 6 in the main paper includes the supervised GAR loss and the unsupervised MCCL loss, the 2 clips will be put to different clusters and pushed further as training goes.

Key Insight The key novelty of **COMPOSER** is that the model learns consistent multiscale representations. The idea of considering scales as views and encouraging scale agreement is applicable to numerous Computer Vision tasks, including general-purpose visual model pre-training, because the entities and scales we have considered are common in human-centered videos. By design, **COMPOSER** is capable of modeling multi-actor multi-object interactions in images or videos. Moreover, **COMPOSER** offers numerous useful practices, including auxiliary prediction to aid training stacks of Transformers, and techniques that can aid the model to learn the high-level knowledge from the low-level coordinate-based keypoint signals (e.g. data augmentations with random perturbation and OKS-based keypoint features to mitigate the issue of noisy estimated keypoints).

Limitation As shown in the failure cases of **COMPOSER**, videos with severe occlusions remain challenging. Severe occlusion can be a limitation for all GAR

methods as their modalities are derived from the RGB input. **COMPOSER** might handle occlusion better than RGB-based methods. In partial occlusion scenarios (i.e., only a ratio of keypoints are occluded for the person – examples are cases shown in Fig. 6 of the main paper, Fig. 7 and Fig. 9 in the Appendix), because **COMPOSER** learns human motion dynamics, better representation of occluded persons can be inferred from the keypoints. In addition, **COMPOSER** is agnostic to the keypoint backbone, and many SOTA keypoint extractors are robust to occlusion (e.g. BlazePose [?,2]).

Societal Impact Human group activity recognition has widespread societal implications in a variety of domains including security, surveillance, kinesiology, sports analysis, and rehabilitation. Privacy and ethical concerns might be raised when deployed in real-world settings if not done in a careful manner. In response to these concerns, **COMPOSER** utilizes only keypoint input and does not use any personally identifiable information for inferring group activity. Even for the backbone which **COMPOSER** is agnostic to, there are existing works [39] that perform privacy-preserving pose estimation. Hence, our method can prevent the sensor camera from acquiring detailed visual data that may contain private or biased information of users.

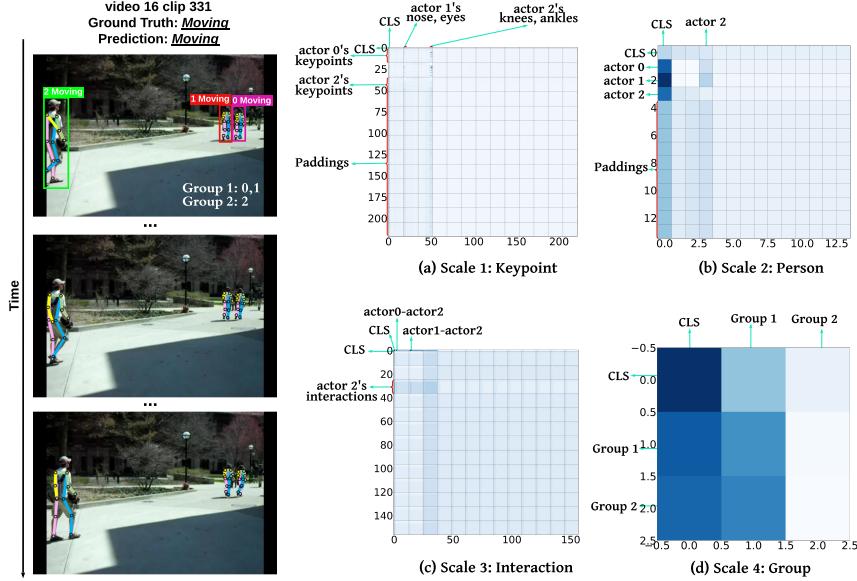


Fig. 5. Qualitative results of COMPOSER on CAD – showcasing attention matrices of a test set instance in the “moving” class.

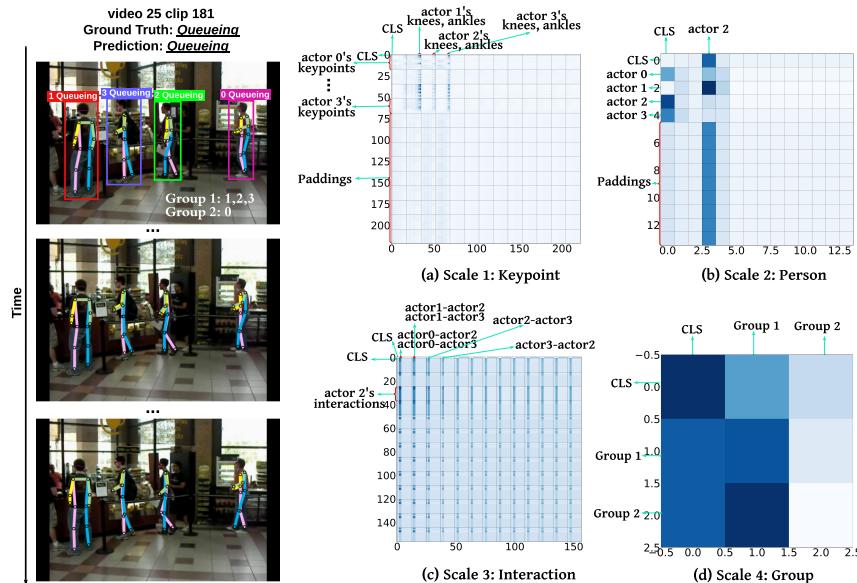


Fig. 6. Qualitative results of COMPOSER on CAD – showcasing attention matrices of a test set instance in the “queueing” class.

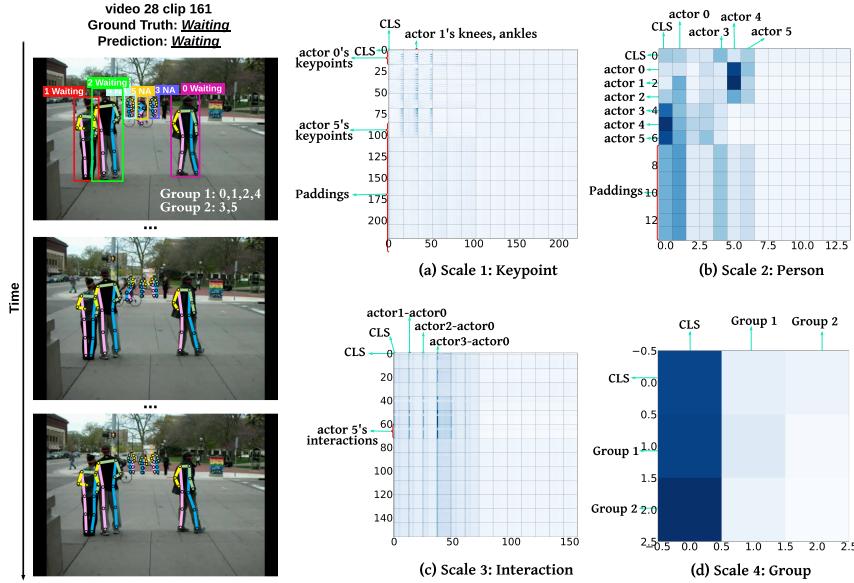


Fig. 7. Qualitative results of COMPOSER on CAD – showcasing attention matrices of a test set instance in the “waiting” class. It is noteworthy that the keypoints of actor 4 and 5 are noisy due to occlusion; yet COMPOSER makes a correct prediction.

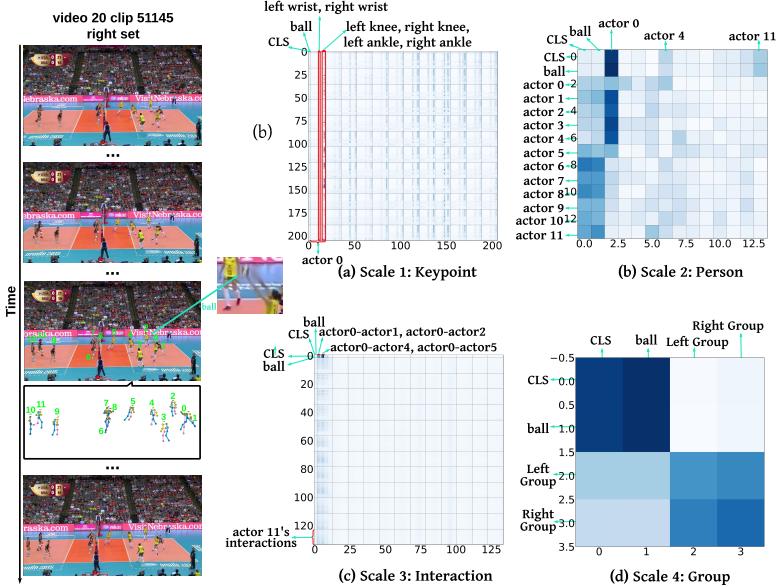


Fig. 8. Qualitative results of COMPOSER on VD – showcasing attention matrices of a test set instance in the “right set” class (key actor is actor 0).

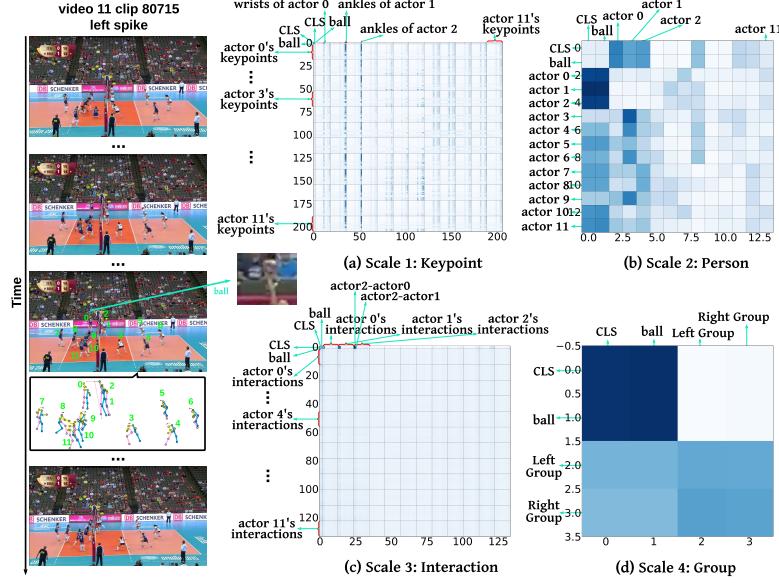


Fig. 9. Qualitative results of COMPOSER on VD – showcasing attention matrices of a test set instance in the “left spike” class (key actor is actor 0).

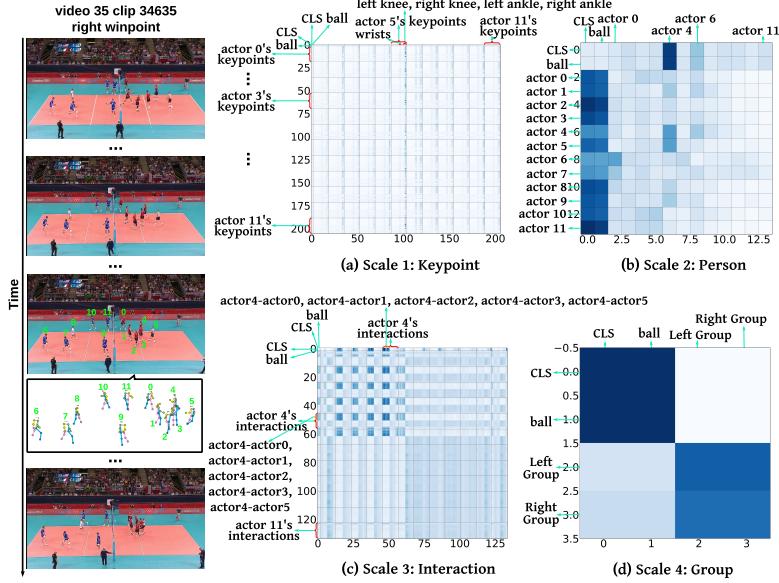


Fig. 10. Qualitative results of COMPOSER on VD – showcasing attention matrices of a test set instance in the “right winpoint” class.

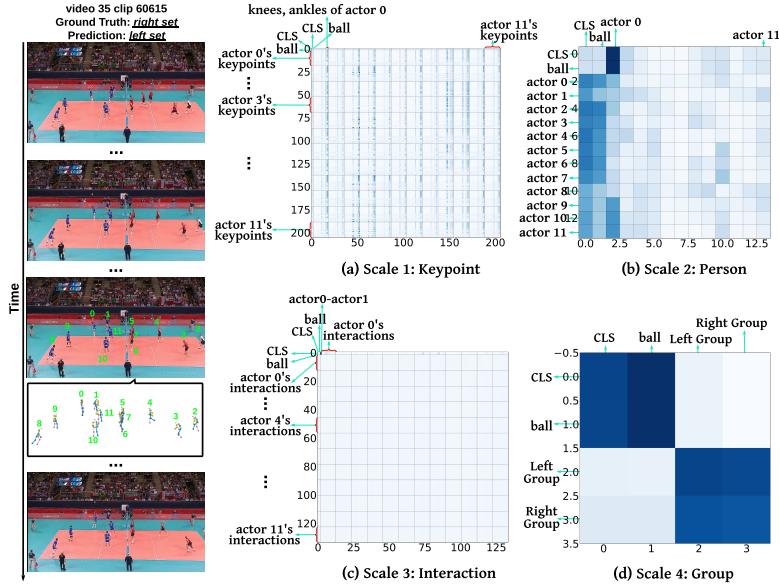


Fig. 11. Mislabeled test set clip example of VD. The annotated label is wrong but the prediction from COMPOSER is correct (key actor is actor 0).

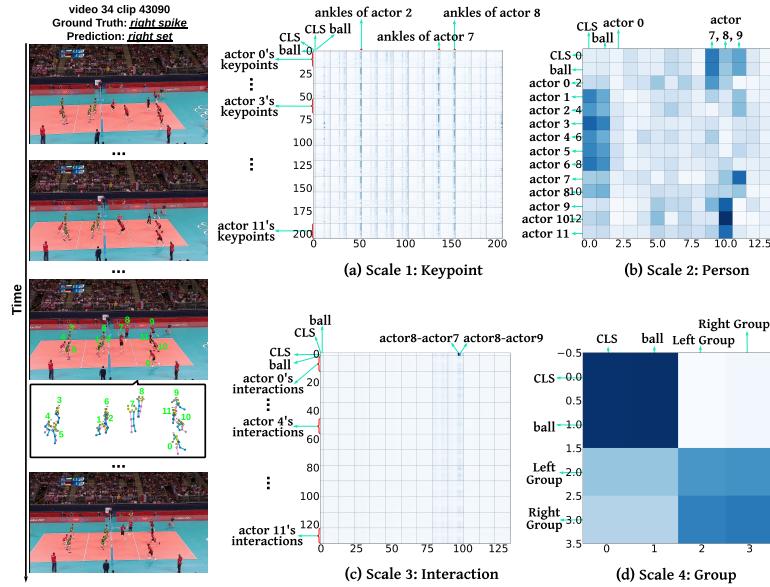


Fig. 12. Mislabeled test set clip example of VD. The annotated label is wrong but the prediction from COMPOSER is correct. Actor 8 is performing the key action *setting* and interacting with the ball. Actor 7 is performing an action very similar to *spiking* but actor 7 is missing interaction with the ball.

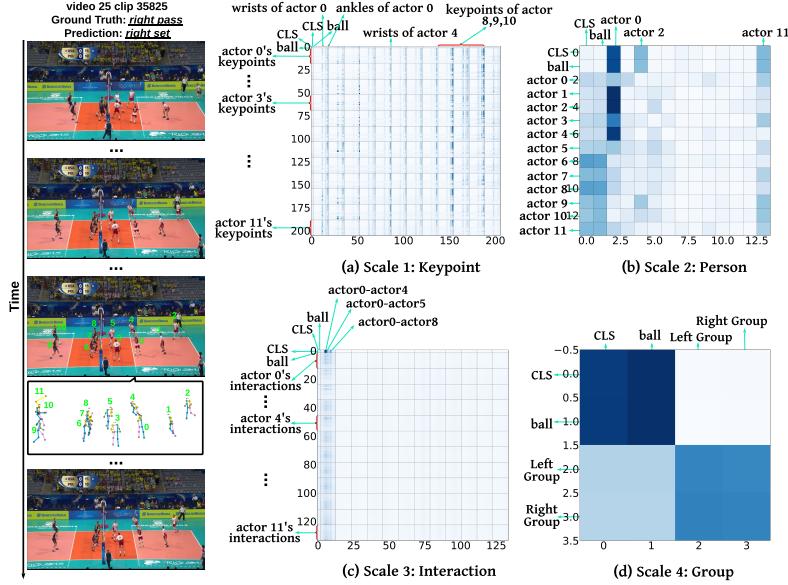


Fig. 13. A failure case on VD. Actor 0 is the one performing the key action. At scale 2 and 3, COMPOSER successfully identifies tokens associated with the key person as the most important tokens. However, at scale 1, COMPOSER fails to focus on keypoints of actor 0 and eventually makes a wrong group activity prediction.

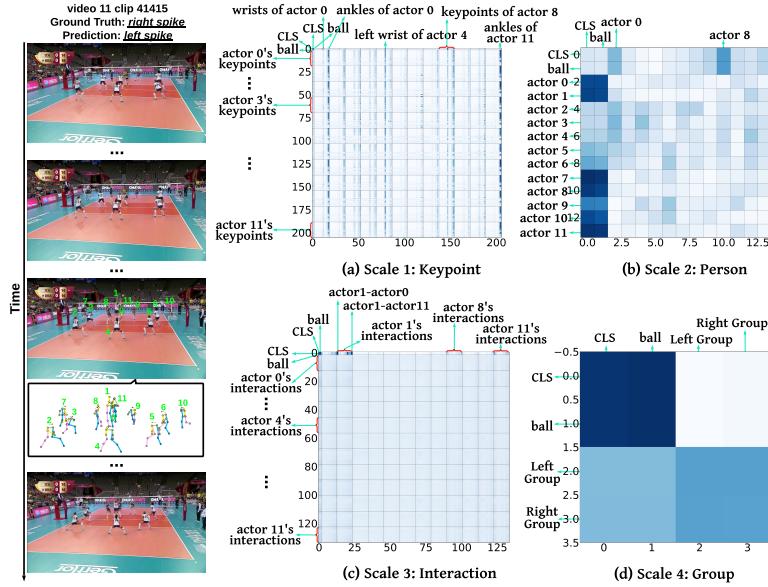


Fig. 14. A failure case on VD. Camera position of this clip is different from other clips in VD, and COMPOSER fails to distinguish which team performs the activity.

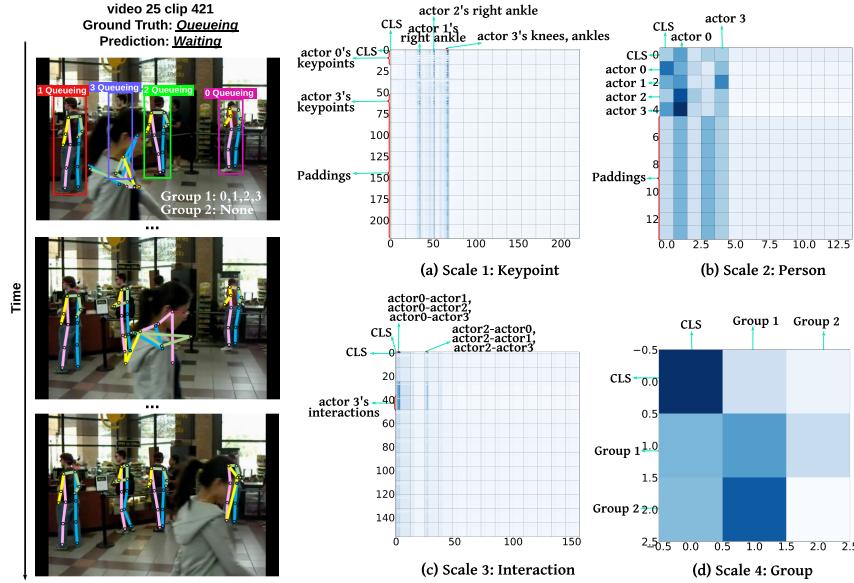


Fig. 15. A failure case on CAD. Because of the severe occlusion, the person queue is interrupted and COMPOSER predicts *waiting* instead of *queueing*.

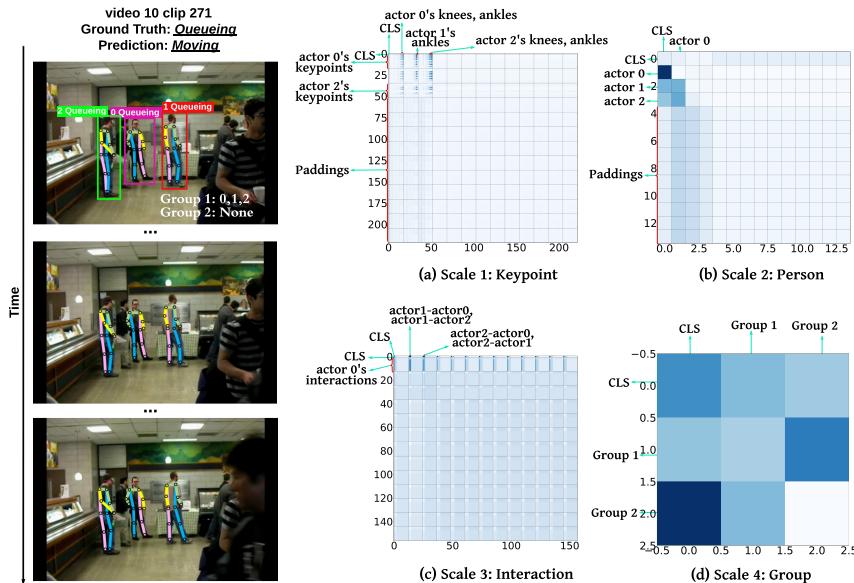


Fig. 16. A failure case on CAD. The movement of actor 1's leg might cause the wrong prediction of COMPOSER.

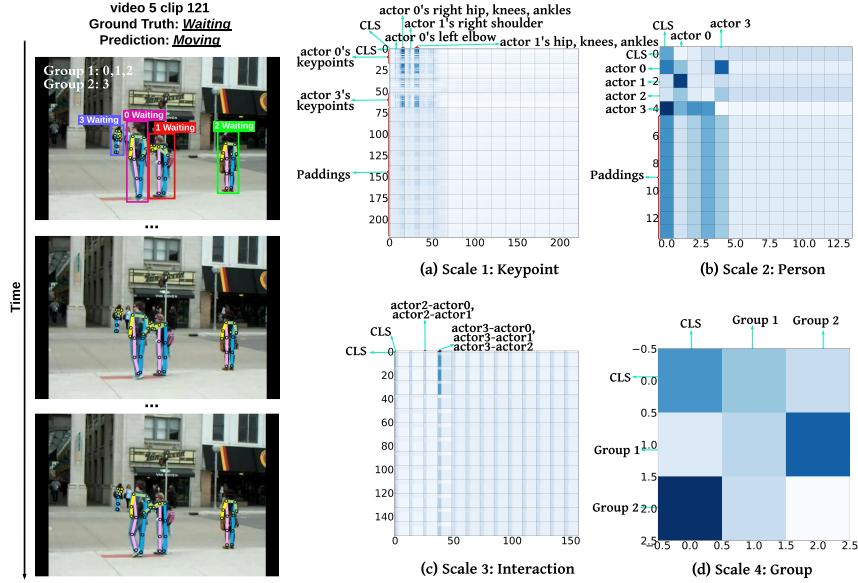


Fig. 17. A failure case on CAD. The movement of actor 0's leg might cause the wrong prediction of COMPOSER.

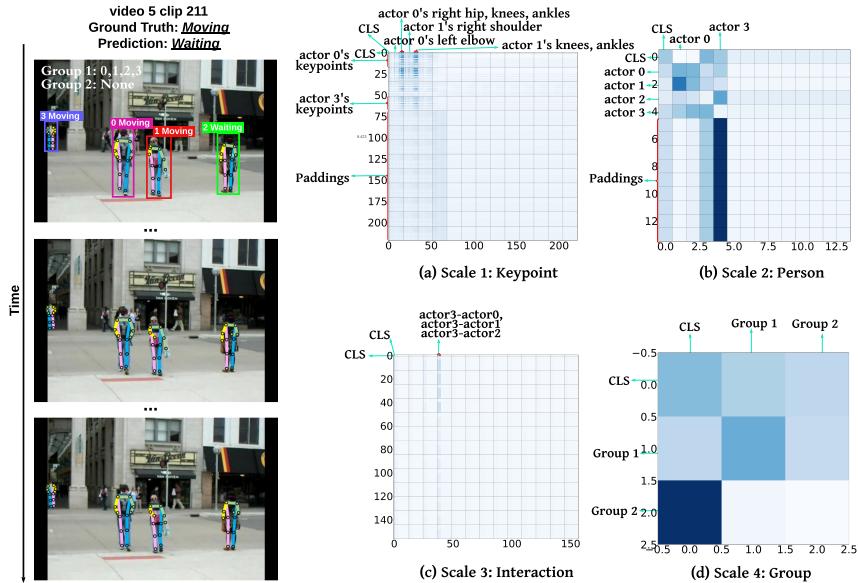


Fig. 18. A failure case on CAD. In the clip, persons were waiting before starting to cross the street. The dynamics are not easily perceptible due to the short temporal window of the clip.