Learning from Synthetic Human Group Activities
Supplementary Material

This document provides more details about our data generator, datasets, and experiments. We also include additional results to supplement the main text. In addition to this supplementary material, all the data, code, videos, tools and related materials are organized at http://cjerry1243.github.io/M3Act.

A. Data Generator: M^{3}Act

A.1. Authoring of Group Activities

Authoring group activities in M^{3}Act is non-trivial because people adhere to social norms while forming groups. The authoring requires nuanced adjustments varied from group to group, including the alignment of characters, their orientations, and the permitted atomic actions. We summarize these rules and adjustments in Tab. 1. For example, characters in a talking group are positioned in a circle facing the center. For queueing groups, characters can form a straight line, curve, etc., and individuals in a queue can be texting, idling, talking, and so on.

Group walking, running (jogging), and dancing are the three group activities with drastic body movements. Particularly, collision avoidance is one social norm that is implicitly followed by humans during collective walking or running. Therefore, we propose a simple algorithm to dynamically adjust the animation speed for each character engaged in the walking and running activities, thus mitigating avatar collisions. The detailed algorithm is shown in Alg. 1. Specifically, the animation speed of a character decreases when any character is in front of it and close to it. When no potential collision is detected, the animation speed of a character can increase up to its initial speed. While animating complex interactions for group dancing is challenging, we enforce nearly synchronous movements for all individuals.

Algorithm 1 Dynamic speed adjustment for collision avoidance.

Require: a list of instantiated characters and their initial animation speeds.

At every frame,
for *character* in Characters **do**
  init_speed ← Initial Speed of *character*
  Speed ← Current Speed of *character*
  pos ← Position of *character*
  forward ← Forward Vector of *character*
  flag ← False
  for *other_character* in Characters **do**
    if *character* is not *other_character* **then**
      pos_other ← Position of *other_character*
      offset ← *pos_other* - *pos*
      angle ← Angle Between *forward* and *offset*
      dist ← Length of *offset*
      if *dist* $\leq$ 0.8 & *angle* $\leq$ 60 **then**
        flag ← True
      end if
    end if
  end for
  if *flag* **then**
    Speed ← Max(*Speed* * 0.96, 0.1)
  else
    Speed ← Min(*Speed* * 1.03, *init_speed*)
  end if
end for

A.2. List of Variables for Domain Randomization

Domain Randomization allows M^{3}Act to generate massive-scale diverse group activity data. Compared to People-SansPeople [11], M^{3}Act contains a much higher degree of domain randomization for animating human motions and activities. M^{3}Act consists of a total of 14 atomic action classes and 384 animation clips, each with several blended style parameters such as character arm-space and stride. The domain randomization covers the scenes, cameras, lights, multi-groups, and activity authoring, as listed in Tab. 2. We describe the randomizers in M^{3}Act below.

- **Scene Selection Randomizer** randomizes the selection of 3D scene.
- **HDRI Randomizer** randomizes the selection of panorama HDRIs.
- **Camera Position Randomizer** includes the randomizations of camera height, distance, and angle in a cylindrical coordinate.
- **Light Type Randomizer** randomizes the light type.
- **Light Position Randomizer** randomizes the positions of all lights.
- **Light Intensity Randomizer** randomizes the intensities of all lights.

Activity Name	Alignment	Face At	Atomic Actions	Other Conditions
Walking	Straight Line, Circle, Rectangle	Same Direction	walk	Adjust animation speed at runtime
Waiting	Multi-Row Straight Lines	Same Direction	idle, text, talk, point, wave	N/A
Queueing	Straight Line, One-Corner Line, Two-Corner Line, Parabola, Curve	Front of Queue	idle, text, talk, point	N/A
Talking	Circle	Group Center	talk	N/A
Dancing	Multi-Row Straight Lines	Same Direction	dance	Nearly-synchronous movements
Jogging	Straight Line, Circle, Rectangle	Same Direction	run	Adjust animation speed at runtime

Table 1. Rules and adjustments for activity authoring.

Category	Randomizer	Variable	Distribution
Scenes	Scene Selection	scene	a set of prebuilt 3D environments
	HDRI Selection	hdri	a set of collected HDRIs
Camera	Camera Position	radius camera rotation camera height perturbation	Uniform(6, 10) Uniform(0, 360) Uniform(1, 5) Cartesian[Uniform(-1, 1), 0 Uniform(-1, 1)]
Lights	Lighting Volume	volume	a set of lighting volume conditions
	Light Type	light type	a set of light types
	Light Position	XZ position height	Cartesian[Uniform(-20, 20), 0, Uniform(-20, 20)] Uniform(5, 10)
	Light Intensity	intensity range	Uniform(0.5, 3)
	Light Rotation	orientation range	Face at Cartesian[Uniform(-50, 50), 0 Uniform(-50, 50)]
Multi-Group	Group Number	group number range	UniformRange(1, MaxNumGroups)
	Group Selection	group activity	a set of modular groups
	Group Placement	group position group rotation	Cartesian[Uniform(-20, 20), 0, Uniform(-20, 20)] Euler[0, Uniform(0, 360), 0]
Activity Authoring	Character Number	character number range	UniformRange(1, MaxNumCharacters)
	Multi-person Subgroup	multi-person number range	UniformRange(0, MaxNum)
	Character Selection	character	a set of 2200 characters
	Character Texture	body color clothes colors	RGBA[Uniform(0.4, 1), Uniform(0.4, 1), Uniform(0.4, 1), Uniform(0.6, 1)] HSV[Uniform(0, 1), Uniform(0, 1), Uniform(0.4, 1)]
	Character Alignment	alignment method	a set of aligment methods
	Character Interval	interval	Uniform(MinInterval, MaxInterval)
	Character Perturbation	position perturbation rotation perturbation	Uniform(-0.25*interval, 0.25*interval) Uniform(-45°, 45°)
	Atomic Action	atomic action	a set of permitted atomic actions
	Animation	animation clip blended parameter speed normalized starting time	a set of animation clips Uniform(0, 1) Uniform(0.8, 1.2) Uniform(0, 1)

Table 2. List of randomizers, variables, and distributions used in $\mathrm{M}^3\mathtt{Act}$.

- **Light Rotation Randomizer** randomizes the rotations of all lights.
- **Group Number Randomizer** randomizes the number of groups being instantiated during the simulation.
- **Group Selection Randomizer** randomizes the selection of the activity for each group in the scene.
- **Group Placement Randomizer** randomizes the center position for each group.
- **Character Number Randomizer** randomizes the number of characters being instantiated in a group.
- **Multi-person Subgroup Randomizer** randomizes the number of *subgroups* in an activity, such as two persons talking to each other in a queueing group. This randomizer applies to queueing and waiting groups.
- **Character Selection Randomizer** randomizes the selection of characters.
- **Character Texture Randomizer** randomizes the clothes and body colors of all characters.
- **Character Alignment Randomizer** randomizes the method used to align characters in a group.
- **Character Interval Randomizer** randomizes the interval between characters.
- **Character Perturbation Randomizer** adds small perturbations to the characters' positions and rotations.
- **Atomic Action Randomizer** randomizes the selection of permitted atomic actions.

Dataset	MOT17 [19]	DanceTrack [21]	M^{3}ActRGB†
#Videos	14	100	**2500**
Avg. #Tracks	**56**	9	8
Avg. Track Len. (s)	35.4	**52.9**	4.7
FPS (s)	30	20	20
Total Frames	11,235	105,855	**250,000**

Table 3. Comparison of multi-object tracking datasets. M^{3}ActRGB† consists of "WalkRun" and "Dance" data used in our tracking experiments.

- **Animation Randomizer** includes the randomization of animation clips, blended style parameters, animation speeds, and playback offsets.

B. Datasets

M^{3}ActRGB contains 9K videos of multi-group and 6K videos of single-group activities, with a total of 6M RGB images and 48M bounding boxes. We show a collage of images from M^{3}ActRGB in Fig. 4, which contains diverse and realistic multi-group and single-group activities. (See our supplementary video for animated data samples.) The distribution of M^{3}ActRGB is also shown in Fig. 1. M^{3}ActRGB contains as many as 19 persons per frame and an average of 8.1 persons per frame. Additionally, we show the comparison of several tracking datasets in Tab. 3. Even though we only selected the "WalkRun" and "Dance" data from M^{3}ActRGB for the tracking experiments, the dataset size is much larger compared to MOT17 [19] and DanceTrack [21]. In terms of the average number of tracks per video, our dataset is closer to DanceTrack. MOT17 mostly contains crowded scenes, while DanceTrack has only one dancing group per video.

M^{3}Act3D has 65K simulations of 3D single-group motions with a total duration of 87.6 hours, captured in 30 FPS. Unlike M^{3}ActRGB which contains equally simulated group activities, M^{3}Act3D has different data sizes of all semantic groups based on their complexity. The complexity of a group includes its alignment methods, permitted atomic actions, animation clips, and styles. Fig. 2 shows the distribution of M^{3}Act3D. Specifically, M^{3}Act3D has more queueing groups than talking groups because the persons can form various shapes and more atomic actions can be performed within a queueing group. We also slightly increased the range of the number of persons in the group for M^{3}Act3D. On average, it has 6.7 persons for every single group and a maximum of 27 persons.

C. More Details of Experiments

C.1 Multi-Person Tracking

The goal of multi-person tracking (MPT) is to predict the trajectories (bounding boxes + identification) of all persons

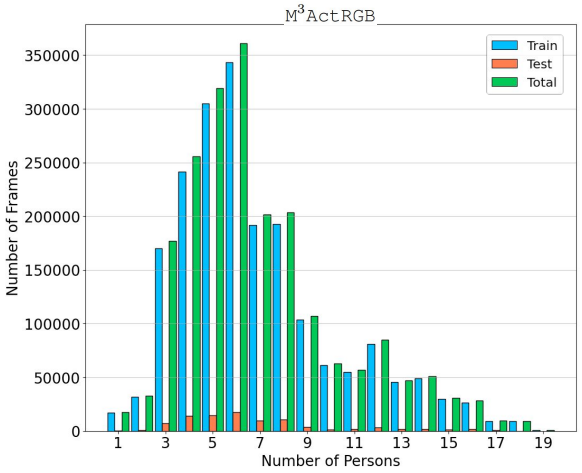

Figure 1. Distributions of M^{3}ActRGB.

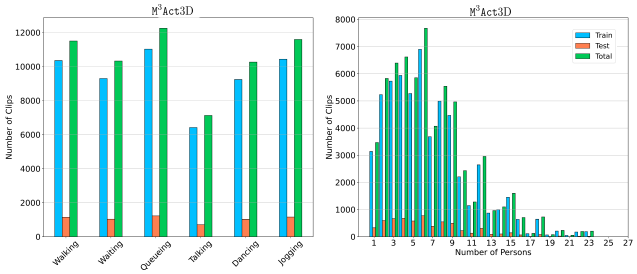

Figure 2. Distributions of M^{3}Act3D.

across an image sequence from a dynamic video stream. Traditionally, multi-object tracking is approached by adding a re-identification layer, either using trainable architecture [26, 27] or applying heuristics-based algorithms [2], on top of the object detection results, aiming to associate the bounding boxes across frames. Recently, end-to-end methods [31, 33] have shown to be more effective in several challenging datasets, such as DanceTrack [21] and MOT17 [19]. To demonstrate the effectiveness of our synthetic data in enhancing real-world performance in multi-person tracking, we assess the impact primarily on MOTRv2 [33]. It is an extension of MOTR [31] by incorporating YOLO-X [14] for bootstrapping detections. Using an end-to-end benchmark allows us to evaluate improvements with the synthetic dataset in both detection and identification.

Implementation Details. We follow the same hyperparameters and data preprocessing procedure from the author-provided MOTRv2 repository[1] for all training jobs. For mixed training with our synthetic data, we simply combined both data from M^{3}ActRGB and DanceTrack as one large dataset, without any additional probability sampling from the real and synthetic data. All models were trained with

[1] https://github.com/megvii-research/MOTRv2

16 NVIDIA A4000 GPUs, using a batch size of 1. It took roughly 7 days of training for all synthetic and real data combined.

C.2 Group Activity Recognition

Understanding collective human behaviors and social groups brings significant importance to various domains, including humanoid robots, autonomous vehicles, and human-computer interactions [4–8, 12, 18, 28]. State-of-the-art methods for group activity recognition (GAR) leverage 2D skeletons as input due to the effectiveness and robustness gained from the less biased and more action-focused representations [13, 20, 23, 30, 32, 34]. We describe the details of 2D skeleton-based GAR experiments that were primarily studied in our main paper. Let $[s_1, \cdots, s_t]$ denote a video with t frames and n persons, each frame $s_i := \{p_1, \cdots, p_n\}$ where $p_i := [(x_1, y_1, c_1), \cdots, (x_j, y_j, c_j)]$. Here, j represents the number of joints in a person's skeleton, and each three-tuple (x_j, y_j, c_j) respectively denotes the x and y coordinates in the pixel space and the class c of the joint. respectively denotes the x, y coordinates in the pixel space and the class c of the joint. For the given input $[s_1, \cdots, s_t]$, the objective of GAR is to output the class of the group activity performed by the dominant group among these n persons and identify the action class of each individual in the video. Usually, the task assumes each video has one dominant group, as any outlier person does not contribute to the group.

Implementation Details. Our implementations of Composer [34] and Actor Transformer [13] for the experiments are based on the open-sourced implementation and hyperparameter settings[2]. The only modified hyperparameter is the batch size, from 384 to 256 due to our computation constraints with an NVIDIA RTX 3090 graphics card. Both Composer and Actor Transformer are transformer [25] based architectures. Note that both architectures are slightly modified after synthetic pre-training, due to the differences between the synthetic and real datasets in the maximum number of persons in a clip and the number of atomic action classes. Specifically, we set a different maximum sequence length to the transformer encoders of Composer and Actor Transformer and replaced the last layers (i.e., the group activity classifier and the person action classifier) with new classifiers to output the correct data shape for the target real-world dataset.

C.3 Controllable 3D Group Activity Generation

Let $G^{TP} = \{m^{in}\}_{i=1\sim T, n=1\sim P}$ be a group of human motions with a total of T frames and P persons. The 3D pose of each person is denoted as $m^{in} \in R^{j\times d}$, where j is the number of joints of a person and d is the joint's

[2] https://github.com/hongluzhou/composer

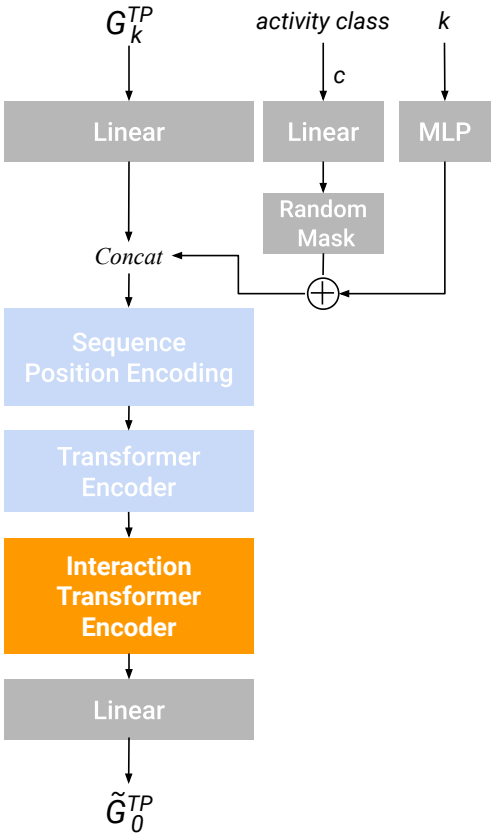

Figure 3. The architecture of MDM+IFormer. The model takes the noised group motion G_k^{TP}, an activity class, and a time step k as inputs and outputs $\tilde{G}_0^{TP}$, an estimation of the clean group motion. The interaction transformer encoder is "added" after the transformer encoder layer of MDM [22] for modeling the person interactions.

feature dimension. We concatenate the global root position and all joints' 6D rotations [6, 35] as the pose representation. Therefore, $j = 26$ (with 25 actual joints) and $d = 6$. The same representation is used as the input for 3D group activity recognition with Composer [34] and used as the ground truth during training for both MDM [22] and MDM+IFormer baselines for 3D group activity generation.

Composer for 3D Group Activity Recognition. The learning-based metrics (recognition accuracy, FID, diversity, and multimodality) [15] that are used to evaluate the generated 3D group activities require a well-trained 3D group activity *recognition* model. Composer [34] is a 2D skeleton-based group activity recognition model with a multi-scale transformer-based architecture. We chose Composer because it is a hierarchical architecture with fine-grained latent group-level and person-level features. We modified the first layer of Composer so that it accepts as input the aforementioned 3D group representation. We used the same set of hyperparameters as the GAR experiments including a learning rate of 0.0005, a weight decay of 0.001,

a hidden dimension of 256 for the transformer encoders, except a batch size of 64, a maximum number of 27 persons, and a total number of 26 joints.

After the 3D group activity recognition model is well-trained, we obtain the 3D group activity recognition accuracy, and extract latent group and person features to calculate the FID, diversity, and multi-modality metrics. The latent group representation is the learned CLS token from the *last* block and last scale of the Multi-Scale Transformer module of Composer, whereas the latent person representations are the learned person tokens from the *last* block and the second scale (i.e., the transformer encoder of the person scale) of the Multi-Scale Transformer module.

MDM & MDM+IFormer. We follow the same diffusion scheme as MDM [22] to obtain the noised group activity at every k-th diffusion time step, G_k^{TP}. Specifically, $G_0^{TP} := G^{TP}$, meaning no noise is added at the 0-th diffusion time step. The reversed diffusion process is then formulated as:

$$\tilde{G}_0^{TP} = D(G_k^{TP}, k, c), \tag{1}$$

where D is the MDM+IFormer network, illustrated in Fig. 3. $\tilde{G}_0^{TP}$ is the estimated clean group activity and c is the one-hot activity label. The loss function follows the objective in [17] and is defined as:

$$L = \left|\left| G_0^{TP} - \tilde{G}_0^{TP} \right|\right|^2. \tag{2}$$

Our implementations of the MDM and MDM+IFormer baselines for 3D group activity generation are based on the author-released implementation of MDM[3] without any hyperparameter tuning. Both models were optimized using the same loss function described above and trained on an NVIDIA RTX 3090 graphics card for 320K iterations.

Formulas of Social Repulsive Forces [16]. The three position-based metrics (in Sec. 4.3.1 of the main paper) are formulated as follows.

– *Repulsive interaction force:*

$$\vec{f}_{ij}^{int} = A \cdot exp[(r_i + r_j - d_{ij})/B] \cdot \vec{n}_{ij}. \tag{3}$$

$\vec{f}_{ij}^{int}$ is the interaction force of character j applied to character i. A and B are constants ($A := 2,000$ and $B := 0.08$). r_i and r_j are the radius of the characters. d_{ij} is the distance between the characters. $\vec{n}_{ij}$ is the unit vector pointing from character j to character i.

– *Contact repulsive force:*

$$\vec{f}_{ij}^{cont} = k \cdot max(0, r_i + r_j - d_{ij}) \cdot \vec{n}_{ij}, \tag{4}$$

where k is a constant ($k := 120,000$).

[3] https://github.com/GuyTevet/motion-diffusion-model

#Epochs required for model convergence (Composer [34] / Actor Transformer [13])		Target CAD2 [10]
Source	CAD2 [10]	88 / 233
	M^{3}Act	13 / 92

Table 4. Pre-training with our synthetic data leads to faster model convergence on the target domain for GAR. (Composer: 6.8× faster; Actor Transformer: 2.5× faster)

– *Total repulsive force:*

$$\vec{f}_{ij}^{total} = \vec{f}_{ij}^{int} + \vec{f}_{ij}^{cont}. \tag{5}$$

All constants follow the social force model proposed in [16].

Generated 3D Group Activities. Please refer to our supplementary video for the rendered 3D group activities generated by MDM and MDM+IFormer.

D. Additional Experiments

Pretraining with data from M^{3}Act can improve convergence speed on the target dataset. We conduct the GAR experiment using 2D skeletons as the only input modality for both models and compare the number of epochs required for model convergence in Tab. 4. To automatically determine whether or not the model training has saturated, we adopted early stopping by setting a maximum number of 500 epochs with stopping patience of 50 epochs. The results suggest that training Composer from scratch on CAD2 requires 88 epochs on average; with M^{3}Act pre-training, Composer only requires 13 epochs for convergence.

E. Limitations & Future Work

We demonstrate that synthetic data can replace a great amount of real data [9] and successfully mitigate the scarcity of real data for multi-person and multi-group tasks, despite the domain gap that restricts the generalizability of models trained with synthetic data. With the release of our data generator, M^{3}Act, we encourage the community to create their own data or enhance the synthetic data generator. While collecting more assets and generating more data with adjusted camera views can surely increase data diversity and shorten the gap, we would also like to point out some aspects of the generator that should be addressed in the future to create more realistic data.

Publicly-available assets. Most assets we use in M^{3}Act are publicly available, including HDRIs, human characters, and animations. However, most existing assets such as photometric 3D scenes and high-quality avatars may require additional licensing for model training. Some assets could be restricted to specific game engines, which hinders the development of synthetic data.

Simulated Hair and clothes. The avatars used in M^{3}Act do not contain hair and clothes physics that are in accordance with their body motions. Adding cloth and strand-based hair simulation would be sufficient for realistic interactions between hair/cloth and body, and thus improve the data quality.

Finger and face Movements. Our animations do not contain finger and face movements. While it might be reasonable as human groups are generally captured from a distance, adding finger and face movements can still improve the fidelity of the human motions.

Human-environment interactions. Like most synthetic datasets [1, 3, 24, 29], M^{3}Act lacks meaningful interactions between human and environment. The interactions might include groups of humans navigating in a complex environment, a person picking up a phone while texting, or holding a suitcase. Animating human motions and activities with scene awareness is incredibly challenging. A simple solution is to polish each scene by carefully placing the avatars and staging the human behaviors. However, it would require significant manual efforts and limit the scale and diversity of the synthetic data.

Complexity for human groups. Animating human groups is significantly more challenging than animating the motions of a single person because the complexity (the number of interactions) increases quadratically as the number of persons increases. We apply relatively simple heuristics when designing rules for authoring human groups, which could only reflect a certain portion of real-world activities. These underlying rules that drive the group motions could also lead to datasets that are less complex than real-world ones, limiting the model generalization on downstream tasks. Creating new groups with expert-guided heuristics, LLM-generated rules, or directly from the 3D GAG method, should be considered. An alternative would be using existing motion capture data in replace of the procedural generation method. However, the lack of fine-grained motion capture data for large-scale collective 3D group motions is an obstacle to the development.

Societal Impacts. While we demonstrate the effectiveness of our synthetic data on several tasks, it is important to note that the use of synthetic data, in all manners, may still result in unbalanced and biased results. We strive to ensure the inclusiveness and fairness of our datasets by incorporating human avatars of all ages, genders, and ethnicities, providing a representative and equitable approach to generating data for responsible advancement in related fields.

References

[1] Eduard Gabriel Bazavan, Andrei Zanfir, Mihai Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Hspace: Synthetic parametric humans animated in complex environments. *arXiv preprint arXiv:2112.12867*, 2021. 6

[2] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 3

[3] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 6

[4] Che-Jui Chang. Transfer learning from monolingual asr to transcription-free cross-lingual voice conversion. *arXiv preprint arXiv:2009.14668*, 2020. 4

[5] Che-Jui Chang and Shyh-Kang Jeng. Acoustic anomaly detection using multilayer neural networks and semantic pointers. *Journal of Information Science & Engineering*, 37(1), 2021.

[6] Che-Jui Chang, Sen Zhang, and Mubbasir Kapadia. The ivi lab entry to the genea challenge 2022–a tacotron2 based method for co-speech gesture generation with locality-constraint attention mechanism. In *Proceedings of the 2022 International Conference on Multimodal Interaction*, pages 784–789, 2022. 4

[7] Che-Jui Chang, Long Zhao, Sen Zhang, and Mubbasir Kapadia. Disentangling audio content and emotion with adaptive instance normalization for expressive facial animation synthesis. *Computer Animation and Virtual Worlds*, 33(3-4): e2076, 2022.

[8] Che-Jui Chang, Samuel S Sohn, Sen Zhang, Rajath Jayashankar, Muhammad Usman, and Mubbasir Kapadia. The importance of multimodal emotion conditioning and affect consistency for embodied conversational agents. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, pages 790–801, 2023. 4

[9] Che-Jui Chang, Danrui Li, Seonghyeon Moon, and Mubbasir Kapadia. On the equivalency, substitutability, and flexibility of synthetic data, 2024. 5

[10] Wongun Choi, Khuram Shahid, and Silvio Savarese. Learning context for collective activity recognition. In *CVPR 2011*, pages 3273–3280. IEEE, 2011. 5

[11] Salehe Erfanian Ebadi, You-Cyuan Jhang, Alex Zook, Saurav Dhakad, Adam Crespi, Pete Parisi, Steven Borkman, Jonathan Hogins, and Sujoy Ganguly. Peoplesanspeople: a synthetic data generator for human-centric computer vision. *arXiv preprint arXiv:2112.09290*, 2021. 1

[12] Mahsa Ehsanpour, Fatemeh Saleh, Silvio Savarese, Ian Reid, and Hamid Rezatofighi. Jrdb-act: A large-scale dataset for spatio-temporal action, social group and activity detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20983–20992, 2022. 4

[13] Kirill Gavrilyuk, Ryan Sanford, Mehrsan Javan, and Cees GM Snoek. Actor-transformers for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 839–848, 2020. 4, 5

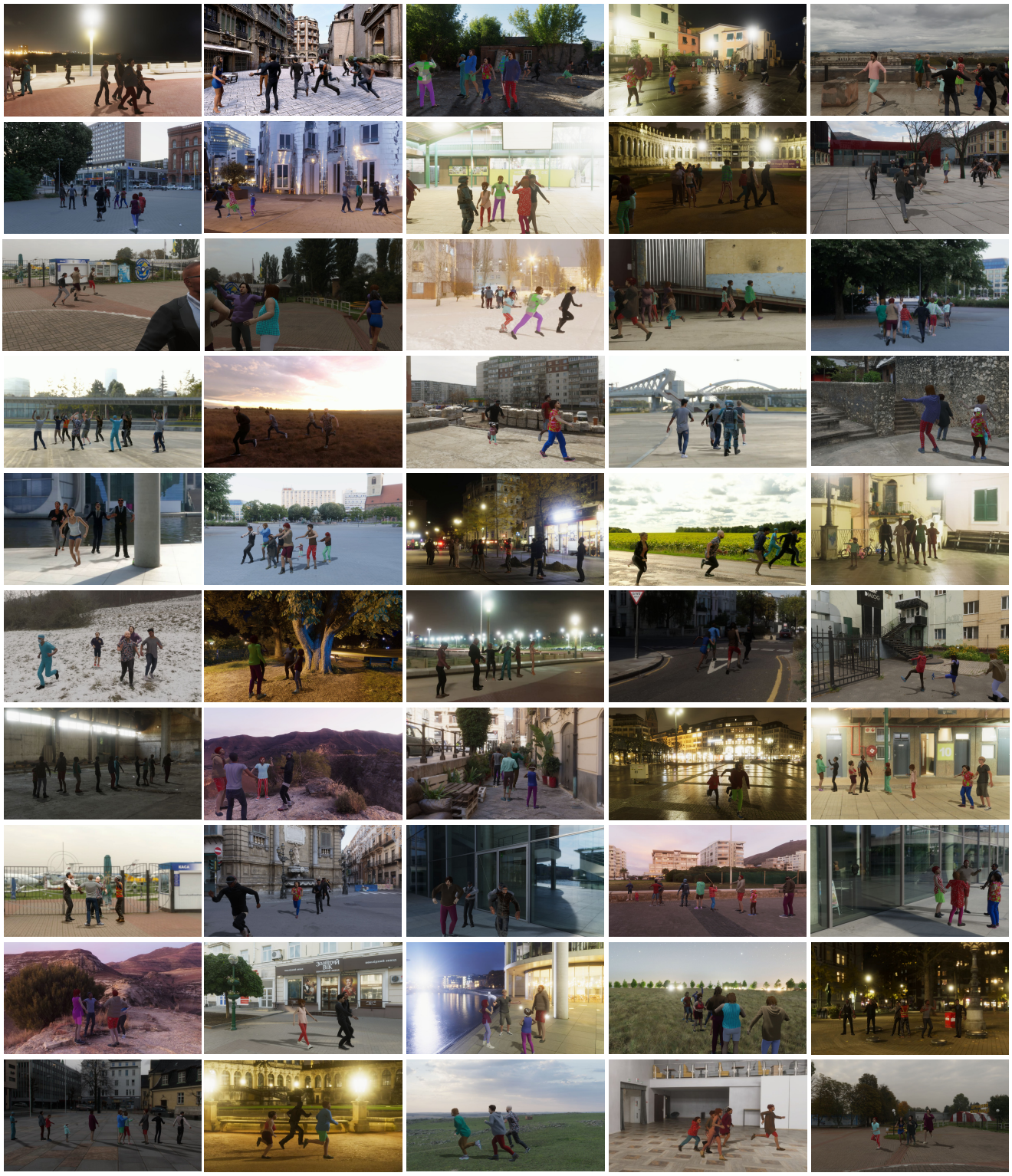

Figure 4. Collage of images from M^{3}ActRGB, including multi-group activities (first 3 rows) and single-group activities (last 7 rows).

[14] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 3

[15] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 4

[16] Dirk Helbing, Illés Farkas, and Tamas Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000. 5

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 5

[18] Roberto Martin-Martin, Mihir Patel, Hamid Rezatofighi, Abhijeet Shenoi, JunYoung Gwak, Eric Frankel, Amir Sadeghian, and Silvio Savarese. Jrdb: A dataset and benchmark of egocentric robot visual perception of humans in built environments. *IEEE transactions on pattern analysis and machine intelligence*, 2021. 4

[19] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 3

[20] Mauricio Perez, Jun Liu, and Alex C Kot. Skeleton-based relational reasoning for group activity analysis. *Pattern Recognition*, 122:108360, 2022. 4

[21] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20993–21002, 2022. 3

[22] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 4, 5

[23] Haritha Thilakarathne, Aiden Nibali, Zhen He, and Stuart Morgan. Pose is all you need: The pose only group activity recognition system (pogars). *Machine Vision and Applications*, 33(6):95, 2022. 4

[24] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 109–117, 2017. 6

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[26] Nicolai Wojke and Alex Bewley. Deep cosine metric learning for person re-identification. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 748–756. IEEE, 2018. 3

[27] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 3

[28] Li-Fang Wu, Qi Wang, Meng Jian, Yu Qiao, and Bo-Xuan Zhao. A comprehensive review of group activity recognition in videos. *International Journal of Automation and Computing*, 18:334–350, 2021. 4

[29] Zhitao Yang, Zhongang Cai, Haiyi Mei, Shuai Liu, Zhaoxi Chen, Weiye Xiao, Yukun Wei, Zhongfei Qing, Chen Wei, Bo Dai, et al. Synbody: Synthetic dataset with layered human models for 3d human perception and modeling. *arXiv preprint arXiv:2303.17368*, 2023. 6

[30] Fabio Zappardino, Tiberio Uricchio, Lorenzo Seidenari, and Alberto Del Bimbo. Learning group activities from skeletons without individual action labels. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10412–10417. IEEE, 2021. 4

[31] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, pages 659–675. Springer, 2022. 3

[32] Xiaolin Zhai, Zhengxi Hu, Dingye Yang, Lei Zhou, and Jingtai Liu. Spatial temporal network for image and skeleton based group activity recognition. In *Proceedings of the Asian Conference on Computer Vision*, pages 20–38, 2022. 4

[33] Yuang Zhang, Tiancai Wang, and Xiangyu Zhang. Motrv2: Bootstrapping end-to-end multi-object tracking by pretrained object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22056–22065, 2023. 3

[34] Honglu Zhou, Asim Kadav, Aviv Shamsian, Shijie Geng, Farley Lai, Long Zhao, Ting Liu, Mubbasir Kapadia, and Hans Peter Graf. Composer: Compositional reasoning of group activity in videos with keypoint-only modality. *Proceedings of the 17th European Conference on Computer Vision (ECCV 2022)*, 2022. 4, 5

[35] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 4