Appendix *for* EgoExo-Fitness: Towards Egocentric and Exocentric Full-Body Action Understanding

Yuan-Ming Li[1,3,‡†], Wei-Jin Huang[1,3,†], An-Lan Wang[1,3,†], Ling-An Zeng[3,4], Jing-Ke Meng[1,3,*], and Wei-Shi Zheng[1,2,3,*]

[1] School of Computer Science and Engineering, Sun Yat-sen University, China; [2] Peng Cheng Laboratory, Shenzhen, China; [3] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China; [4] School of Artificial Intelligence, Sun Yat-sen University, China

A1 Appendix

In this Appendix, we will provide more details about data collection and annotations of the proposed EgoExo-Fitness dataset in Sec. A2. After that, we will introduce details about the benchmarks in Sec. A3, including formal definition, implementation, more experiment analysis, and other benchmark tasks. Finally, we provide more dicussinos about comparisons between EgoExo-Fitness and the existing datasets (*i.e.*, Ego4D [9], Ego-Exo4D [10], and other related datasets) in Sec. A4.

A2 More details of EgoExo-Fitness

Recording System. For the two Insta-Go3 cameras, we use 2560×1440 pixel resolution RGB images. For the GoPro camera, we use 1920×1080 pixel resolution RGB images. For the side and front exocentric cameras, we set the resolution to be 1024×576 and 1280×720, respectively. After synchronization, video frames will be extracted with 30 FPS and resized to 456×256.

Participants. We recruited 40 adults (28 males, 12 females) for data collection. Each participant was asked to participate at most nine rounds of recordings.

Action sequences. EgoExo-Fitness records 86 types of fitness action sequences, each containing 3 to 6 continuous fitness actions. Tab. A.2 provides the details of each action sequence.

Annotation tools. We use the popular COIN [20] annotation tool for two-level temporal boundaries. Besides, we develop a web-based annotation tool to collect the annotations of interpretable action judgment. Fig. A.1 introduce the workflow of the annotation process of interpretable action judgment.

Guidance and Technical Keypoints. As discussed in the main body of the paper, we obtain several technical keypoints from the text guidance. We use the

‡: Project lead. †: Equal key contributions. ∗: Corresponding authors.
Emails: {liym266, wanganlan}@mail2.sysu.edu.cn; mengjke@gmail.com; wszheng@ieee.org.

Table A.1: Recorded fitness actions. Abbr.: the abbreviation of the fitness action.

Action types	Abbr.	Action types	Abbr.	Action types	Abbr.
1: Kneeling Push-ups	KPU	5: Shoulder Bridge	SB	10: Jumping Jacks	JJ
2: Push-ups	PU	6: Sit-ups	SU	11: High Knee	HK
3: Kneeling Torso Twist	KTT	7: Leg Reverse Lunge	LRL	12: Clap Jacks	CJ
4: Knee Raise and Abdominal Muscles Contract	KRAMC	8: Leg Lunge with Knee Lift	LLKL		
		9: Sumo Squat	SS		

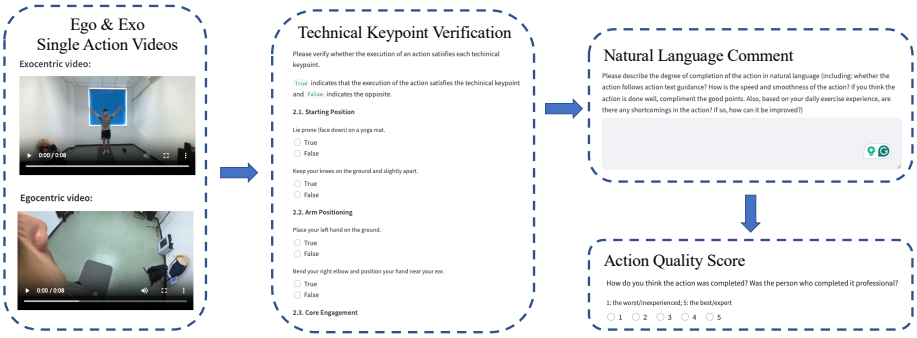

Fig. A.1: The annotation process of interpretable action judgement.

text guidance provided in FLAG3D [21]. We use the prompt "*In this task, you are given text guidance of a fitness action. Your job is to separate the text guidance into several key points.*" to require LLM (*i.e.*, GPT-4) to extract technical keypoints from the text guidance. Tab. A.3 shows an example of the extracted technical keypoints.

More Examples. We show more examples of annotations of interpretable action judgment in Fig. A.2.

Privacy and Ethics. From the onset, privacy and ethics standards were critical to the data collection and release effort. All videos are recorded after we obtain the consent provided by participants. All human experts are asked to sign a privacy protection agreement to prevent data and privacy disclosure during the annotation process. To further protect the privacy and personal information, before the data release, we will ensure that the release resources do not contain privacy-sensitive content (*e.g.*, real names).

Table A.2: All recorded fitness sequences. For the correspondence between action names and abbreviations, please refer to Tab. A.1. SID: sequence ID.

SIDs	Action Orders	SIDs	Action Orders
1	KPU, SU, JJ	44	SB, LRL, CJ, KRAMC
2	KPU, JJ, SU	45	KRAMC, LRL, SB, CJ
3	KTT, SU, JJ	46	PU, SB, CJ, SS
4	KTT, JJ, SU	47	PU, SB, SS, CJ
5	SU, KTT, JJ	48	SB, PU, CJ, SS
6	KPU, JJ, HK	49	PU, SB, LRL, KRAMC, CJ
7	KPU, HK, JJ	50	PU, SB, LRL, CJ, KRAMC
8	JJ, KPU, HK	51	PU, LRL, SB, KRAMC, CJ
9	KPU, SU, LLKL, JJ	52	SB, PU, LRL, CJ, SS
10	KPU, SU, JJ, LLKL	53	PU, SB, LRL, CJ, SS
11	SU, KPU, LLKL, JJ	54	PU, SB, SS, CJ, LRL
12	KTT, SU, LLKL, JJ	55	SB, HK, KRAMC, CJ, SS
13	KTT, SU, JJ, LLKL	56	SB, KTT, KRAMC, CJ, SS
14	LLKL, SU, KTT, JJ	57	SB, KPU, KRAMC, CJ, SS
15	KPU, KTT, JJ, HK	58	SB, CJ, KRAMC, KPU, SS
16	KPU, KTT, HK, JJ	59	SB, CJ, KRAMC, HK, SS
17	KTT, KPU, JJ, HK	60	SB, CJ, KRAMC, KTT, SS
18	KPU, KTT, SU, LLKL, JJ	61	SB, CJ, KRAMC, LLKL, SS
19	KPU, KTT, SU, JJ, LLKL	62	SB, CJ, KRAMC, SU, SS
20	KPU, SU, KTT, LLKL, JJ	63	JJ, SB, KRAMC, CJ, SS
21	KTT, KPU, SU, JJ, HK	64	LLKL, SB, KRAMC, CJ, SS
22	KPU, KTT, SU, JJ, HK	65	PU, SB, LRL, KRAMC, CJ, SS
23	KPU, KTT, HK, JJ, SU	66	PU, SB, LRL, CJ, SS, KRAMC
24	KTT, KRAMC, LLKL, JJ, HK	67	PU, LRL, SB, KRAMC, CJ, SS
25	KTT, JJ, LLKL, KRAMC, HK	68	SB, PU, SU, LRL, CJ, SS
26	KRAMC, KTT, LLKL, JJ, HK	69	SB, PU, LLKL, LRL, CJ, SS
27	KPU, KTT, SU, LLKL, JJ, HK	70	PU, SB, SU, SS, CJ, LRL
28	KPU, KTT, SU, JJ, HK, LLKL	71	PU, SB, JJ, SS, CJ, LRL
29	KPU, SU, KTT, LLKL, JJ, HK	72	PU, SB, LLKL, SS, CJ, LRL
30	KTT, KPU, KRAMC, SU, JJ, HK	73	PU, SB, KTT, SS, CJ, LRL
31	KPU, KTT, KRAMC, SU, JJ, HK	74	SB, HK, LRL, KRAMC, CJ, SS
32	KPU, KTT, KRAMC, HK, JJ, SU	75	SB, CJ, LRL, KRAMC, JJ, SS
33	KTT, KRAMC, SU, LLKL, JJ, HK	76	SU, SB, LRL, KRAMC, CJ, SS
34	KTT, JJ, SU, LLKL, KRAMC, HK	77	LLKL,KRAMC,HK
35	KRAMC, KTT, SU, LLKL, JJ, HK	78	SB,LRL,CJ
36	PU, LRL, CJ	79	PU,SB,CJ,LRL
37	LRL, PU, CJ	80	KTT,SU,JJ,HK
38	SB, CJ, LRL	81	KPU,KTT,SU,JJ
39	LRL, SB, CJ	82	KPU,KTT,LLKL,SU,JJ,HK
40	PU, LRL, KRAMC, CJ	83	KPU,KTT,JJ,HK,LLKL
41	PU, LRL, CJ, KRAMC	84	PU,SB,KRAMC,CJ,SS
42	LRL, PU, KRAMC, CJ	85	KPU,SU,KTT,KRAMC,JJ
43	SB, LRL, KRAMC, CJ	86	KTT,KRAMC,LLKL,SU,JJ,HK

Table A.3: Examples of the language guidance and technical keypoints.

Action Name	Clap-Jacks
Language Guidance	Lift your head and chest, and tense your abdomen. tense the arms, and use the strength of your pectoral muscles to clap your hands while jumping back and forth with alternating feet.
Technical Key Points	**KP_1:** Lift your head and chest upward. **KP_2:** Tense your abdominal muscles for stability. **KP_3:** Keep your arms tense. **KP_4:** Use the strength of your pectoral (chest) muscles. **KP_5:** Clap your hands while performing the exercise. **KP_6:** Perform jumping movements back and forth. **KP_7:** Alternate your feet while jumping.
Action Name	Sumo Squat
Language Guidance	Stand about twice shoulder-width apart, with your toes facing diagonally forward. when squatting to the thighs parallel to the ground, keep your knees in the same direction as your toes. keep your upper body as straight as possible, and sit back slightly when squatting. cross your arms over your chest.
Technical Key Points	**KP_1:** Stand with your feet about twice shoulder-width apart. **KP_2:** Position your toes so they are pointing diagonally forward. **KP_3:** Squat down until your thighs are parallel to the ground. **KP_4:** Ensure your knees are aligned in the same direction as your toes. **KP_5:** Keep your upper body as straight as possible throughout the movement. **KP_6:** Slightly sit back as you squat down, like sitting into a chair. **KP_7:** Cross your arms over your chest.

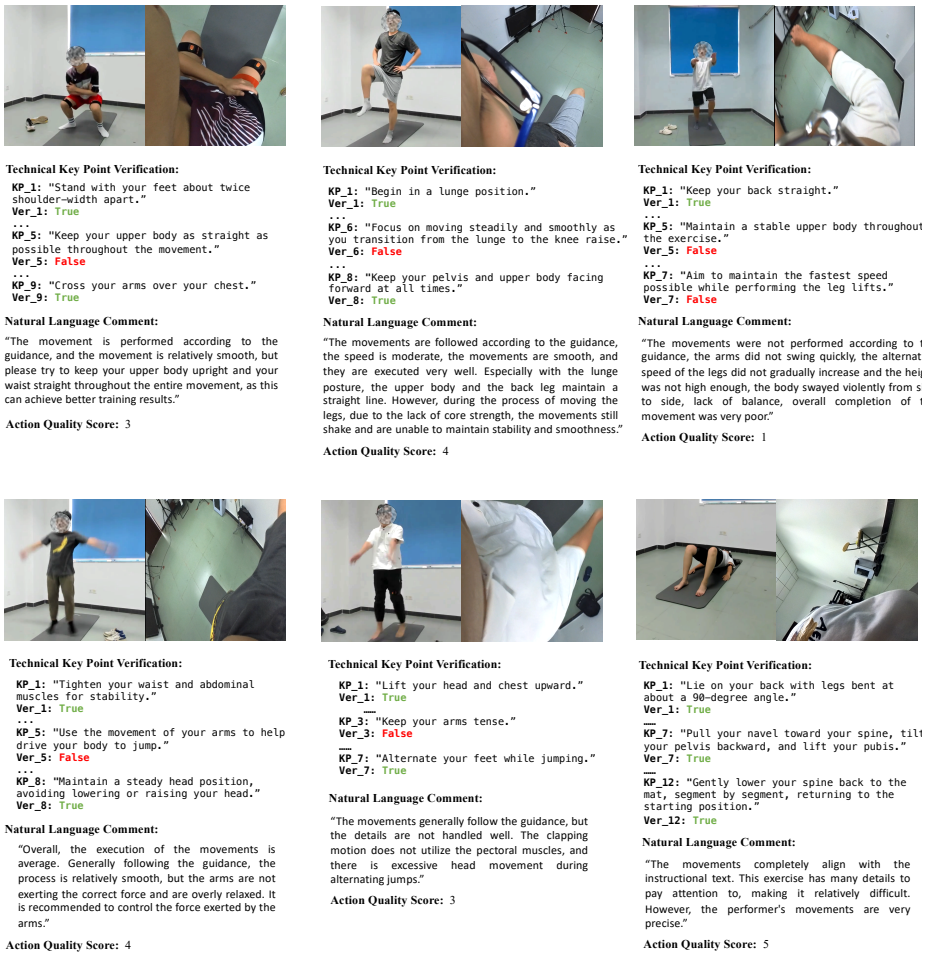

Fig. A.2: More examples of interpretable action judgement.

A3 Benchmarks

In this section, we will first present more details and experiments on Action Classification, Cross-View Sequence Verification, and Guidance-based Execution Verification. Then, we will introduce two more benchmarks on Action Localization and Cross-View Determination.

A3.1 Action Classification

Implementation. (1) **Data Construction**: We select 4,753 single action videos (3,000 for training and 1,753 for testing) to construct the Action Classifica-

tion benchmark. (2) **Pre-trained weights**: We evaluate models with various pre-training strategies to construct action classification benchmark. For I3D [5], EgoVLP [13], and TimeSformer [2] pretrained on the K600 [4] dataset, we use the official pretrained weights. For TimeSformer pre-trained on Ego-Exo4D [10], we follow the setting of "Key-Step Recognition" benchmark in Ego-Exo4D to initialize the model with K600 pre-trained weights then trained on Ego-Exo4D. (3) **Experiment Settings**: The input size of the video clip is set as $16 \times 224 \times 224$. During training, the video clips are sampled with temporal augmentation followed by random cropping. We train the models for 200 epochs with a base learning rate of 1e-5 and adopt a multi-step learning rate decay with a decay rate of 0.5 for every 25 epochs. For evaluation, a single video is uniformly sampled from the video, followed by center cropping.

Table A.4: Action classification results on different views. We report Top-1 accuracies on different veiws for TimeSformer model with Ego-Exo4D pre-training.

Train on	**Exo-L**	**Exo-M**	**Exo-R**	**Exos**	**Ego-L**	**Ego-R**	**Ego-M**	**Egos**
Exo	0.8746	0.8993	0.8746	0.8825	0.0814	0.1017	0.0610	0.0814
Ego	0.1559	0.1475	0.1763	0.1601	0.8305	0.8508	0.7186	0.8000
Ego & Exo	0.9051	0.8921	0.8949	0.8975	0.8475	0.7898	0.7153	0.7840

More Experiment Analysis. In the main paper's experiments, we found that models perform worse on egocentric data. In this section, we will explain these results more fully. The first reason leading to this result is the invisibility of the human body. To support this view, we evaluate the performance of each view. As shown in Tab. A.4, it is more difficult for a model to recognize an action from videos shot from the Ego-M camera (*i.e.*, the forward-recording camera) than from other egocentric cameras (*i.e.*, Ego-L and Ego-R). The main difference between videos shot from Ego-M and other egocentric Ego-cameras is that the human body is always out of view in videos from Ego-M.

Compared with Ego-M, videos shot from Ego-L and Ego-R record parts of the body. However, from Tab. A.4, it can be observed that that the model still achieves poorer performance on videos shot from Ego-L and Ego-R than on those from exocentric cameras. To go deeper to this observation, we conduct a confusion evaluation. Specifically, we select one action (*i.e.*, Leg Reverse Lunge) and two other actions (*i.e.*, Knee Raise and Abdominal Muscles Contract, and Kneeling Torso Twist) whose egocentric videos are much easier to confuse models. The confusion matrixes and cropped frames are shown in Fig. A.3. From the egocentric video frames, similar action patterns (*i.e.*, legs bending) can be observed among videos of these three actions, which cause serious confusion. On the contrary, the exocentric videos of these three actions are much more discriminating, which leads to higher classification performance. From these results, we conclude that the other reason leading to poorer full-body action understanding performance on egocentric videos is that it is easier to observe similar action patterns from egocentric videos, which will confuse models.

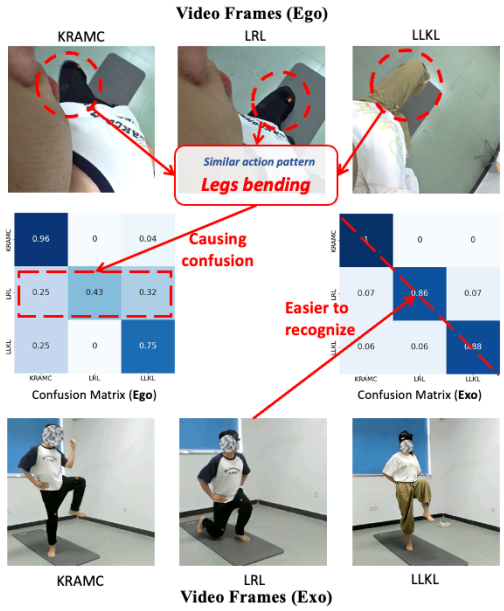

Fig. A.3: The similar action patterns observed by egocentric videos will confuse models to recognize an action. Best viewed in color.

A3.2 Cross-View Sequence Verification

More Details on Task Setup. Following the task setup of existing work on SV [6,17], we formulate CVSV as a classification task during training, *i.e.*, predicting the sequence class. During testing, the embedding distance d (or similarity) between two videos indicates the verification score of this pair.

Specifically, in training phase, a training set $D_{train} = \{(v_i, s_i)\}_{i=1}^{N}$ is used to construct a sequence classification task, where v_i is a action sequence video and s_i is a sequence label (*e.g.*, a SID in Tab. A.2). Given a video $v \in \mathbb{R}^{3\times H\times W\times T}$ and its corresponding sequence label s, the model $f \odot g : R^{3\times H\times W\times T} \rightarrow \mathbb{R}^{C}$ is asked to predict the sequence label from C sequence classes. Here f is the embedding encoder, g is the classifier. H, W, and T are height, width, and the number of frames, respectively. In the testing phase, the model is asked to perform sequence verification on the test set where the sequence labels do not overlap with videos in the training set. Given a video pair (v_i, v_j), a distance (or similarity) function D is conducted on the embeddings of each video in the pair, which is denoted as $d_{ij} = D(f(v_i), f(v_j))$. A higher d_{ij} indicates a lower possibility for v_i to contain the same action sequence as v_j (opposite if similarity function is used). In practical application, a threshold τ can be set to decide whether two sequences are consistent: if $d_{ij} > \tau$, sequences of v_i and v_j are consistent, otherwise inconsistent (opposite if similarity function is used).

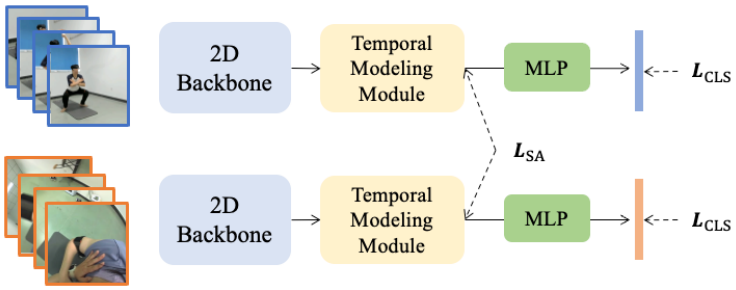

Fig. A.4: An overveiw of the CAT baseline for cross-view sequence verification.

More Details about baseline model. As discussed in the main paper, we adopt the state-of-the-art sequence verification model CAT [17] as the baseline model. The overview of CAT is shown in Fig. A.4. The embedding encoder f includes a 2D Backbone and a Temporal Modeling Module to encode video embeddings. The classifier g is implemented as a Multi-Layer Perceptron. Specifically, the 2D backbone is implemented as a CLIP-ViT/16 [18], and the Transformer encoder is adopted as a Temporal Modeling Module. During training, CAT takes a pair of videos with the same action sequence as input and is optimized to learn to classify the action sequence labels by a classification loss L_{CLS}. Besides, an extra sequence alignment loss L_{SA} is adopted to align video representations of videos with the same action sequence.

Implementation. (1) **Data Construction**. Following previous works [6,11,17], we take 1074 action-sequence videos to build the CVSV dataset and make sure that the type of action sequences in the training set has no overlap with the test set. After that, we select 3,800 video pairs to train CAT and select another 3800 video pairs for testing. (2) **Experiment Settings.** We follow the official setting of existing SV works [6, 11, 17] to use the normalized Euclidean distance is used as the distance function. All experiments are conducted with a batch size of 8, a cosine learning rate scheduler with a base learning rate of 5e-5, and the models are trained for 40 epochs.

A3.3 Guidance-based Execution Verification

More Details about GEVFormer. This section will provide more details on implementing GEVFormer, including the architectures and loss formulation.

In GEVFormer, the TCM module is implemented as a 2-layer Transformer Encoder with 2-head attention. CMV module is designed as a 2-layer Transformer Decoder with 2-head attention and a linear evaluator. The prediction results $P = \{p_1, ..., p_n\}$ is normalized by $Sigmoid(\cdot)$ function.

As discussed in the main paper, two losses are adopted to train GEVFormer (*i.e.*, L_{GEV} and L_{Align}). First, given the predicted results P, the ground-truth targets are denoted as $P^{gt} = \{p_1^{gt}, ..., p_n^{gt}\}$, where p_i^{gt} is a binary value and $p_i^{gt} = 1$ indicates that the execution of the action satisfies the i-th technical keypoint.

Table A.5: Ablation study on different components of GEVFormer.

Methods	TCM	CMV	L_{Align}	Exo F1-score	Exo Precision	Exo Recall	Ego F1-score	Ego Precision	Ego Recall	Avg F1-score
CLIP-GEV				0.5080	0.5362	0.4657	0.4780	0.5401	0.4094	0.4881
	✓			0.5174	0.5407	0.4831	0.5103	0.5492	0.4890	0.5138
	✓	✓		0.5282	0.5502	0.5080	0.5248	0.5570	0.4960	0.5265
GEVFormer	✓	✓	✓	**0.5452**	0.5219	0.5707	**0.5425**	0.5186	0.5687	**0.5439**

After that, L_{GEV} is implemented as a Binary Cross-Entropy loss:

$$L_{GEV} = -\sum_{i=1}[p_i^{gt} log p_i + (1 - p_i^{gt}) log(1 - p_i)]. \tag{A.1}$$

Besides, given a mini-batch of training samples $V = \{v_1, v_2, ..., v_K\}$ (K is the batch size), we randomly sample another batch of video $\widetilde{V} = \{v_1, ...v_K\}$, where v_i and v_j are time-aligned (*i.e.*, synchronized). After that, we fed videos in V and $\widetilde{V}$ into GEVForrmer and get the enhanced visual embeddings (outputs of TCM module), which are denoted as $G = \{g_1, ..., g_K\}$ and $\widetilde{G} = \{\widetilde{g}_1, ..., \widetilde{g}_K\}$, respectively. Given G and $\widetilde{G}$, the synchronized video alignment loss L_{Align} is written as:

$$L_{Align} = \frac{1}{K}\sum_{i=1}^{K} log \frac{exp(\psi(g_i, \widetilde{g}_i)/\delta)}{\sum_{j=1}^{K} exp(\psi(g_i, \widetilde{g}_j)/\delta)}, \tag{A.2}$$

$$\psi(g_i, g_j) = \frac{g_i}{||g_i||} \cdot \frac{g_j}{||g_j||}, \tag{A.3}$$

where $\psi(.,.)$ indicates cosine similarity function, and δ is the tempreture parameter.

More experiment settings. We select 3,260 samples from videos shot by Ego-R, Ego-L, Exo-R and Exo-L. After that, we split them into training set and test set (2,232 videos for training and 1,028 for testing). We use video frames sampled with a sample rate $1/16$ as the input. During training, a random temporal augmentation is used to augment data. By default, λ is set as 0.7.

More Experiment Analysis. In this section, we conduct ablation studies on GEVFormer. We start by ablating the components of GEVFormer. As shown in Tab. A.5, when adding each component from the CLIP-GEV baseline to GEVFormer, performance gradually improved, showing each component's contribution.

A3.4 Action Localization

Task Setups. TAL [15,22,24] aims to identify action instances (*i.e.*, foreground) in time and recognizing their categories. Note that the most discriminating part of Fitness action is the "executing" stage. Hence, in the Action Localization benchmark, we regard an action's "executing" step as the foreground, otherwise

Table A.6: Temporal Action Locaization benchmark results on different views. Results in blue and red indicates the best performance on exocentric and egocentric videos, respectively.

Train on	Test on	AP@0.3	AP@0.4	AP@0.5	AP@0.6	AP@0.7	mAP
Ego	Ego	43.30	41.48	37.61	28.99	16.92	33.66
	Exo	5.32	4.62	3.24	1.89	0.67	3.15
Exo	Ego	4.79	4.05	3.01	1.52	0.61	2.80
	Exo	49.87	48.48	44.78	37.81	23.64	40.92
Ego & Exo	Ego	45.45	43.21	39.29	32.15	18.04	35.63
	Exo	48.47	46.82	43.70	36.22	23.65	39.77

as the background. The model is asked to predict all temporal boundaries and the action type of the foreground given an untrimmed action sequence video containing various actions.

Implementation. (1) Data Construction. We select 1,165 untrimmed action sequence videos and randomly separate them into training and testing sets (66.7% for training and 33.3% for testing). **(2) Baseline Model.** We apply competing state-of-the-art transformer-based TAL method, TadTR [15], using frame-wise features etxtracted from CLIP [18]. **(3) Matrics.** Performance is evaluated by mean average percision (mAP) at different intersections over union (IoU) thresholds of $\{0.3, 0.4, 0.5, 0.6, 0.7\}$. **(4) Other experiment settings.** In our implementation, we use 10 action queries. Following previous work [14, 15], we crop each feature sequence with windows of length 450 and overlap of 75%. We train TadTR on EgoExo-Fitness for 50 epochs with an inital learning rate of 1e-4. For other experiment settings, we follow the official implementation of TadTR [15] on the THUMOS14 dataset [12].

Experiment. The benchmark result on Action Localization is shown in Tab. A.6. In Action Localization, We have similar findings as in the Action Classification benchmark, such as jointly training the model on multi-view data will not benefit localization results on both egocentric and exocentric viewpoints (*i.e.*, only performance on egocentric data achieves improvement).

A3.5 Cross-View Skill Determination

Given a pair of action videos, Skill determination [7, 8] aims at inferring which video displays more skill. Such a task has shown great potential for training humans and intelligent agents. Such a task will benefit the practical application of training humans and intelligent agents. Although previous works have achieved significant progress, today's skill determination dataset is either collected from exocentric viewpoints (*e.g.*, best) or egocentric(-like) viewpoints (*e.g.*, epic-skill). However, in practical application, the videos may come from various viewpoints, which poses a new challenge to skill determination. To address this issue, we extend the traditional skill determination to a cross-view manner (*i.e.*, Cross-View Skill Determination).

Table A.7: Benchmark results on Cross-View Skill Determination. "ego/exo" indicates independent models are trained on ego-only and exo-only data. "ego+exo" indicates that the model is trained on both egocentric and exocentric data.

Methods	Acc		
	Ego-Ego	Exo-Exo	Ego-Exo
Random	0.5000	0.5000	0.5000
RAAN(ego/exo)	**0.7386**	0.7656	-
RAAN(ego+exo)	0.7072	**0.7768**	**0.7241**

Task Setups. Following previous works [7, 8], we formulate cross-view skill determination (CVSD) as a pair-wise ranking task. In this setup, given a video pair (v_i, v_j) where v_i display more skill than v_j, our goal is to learn a ranking function $f(\cdot)$ such that $f(v_i) > f(v_j)$.

Implementation. (1) Data Construction. EgoExo-Fitness provides the action quality scores in annotations of interpretable action judgment. Based on this, we construct the Cross-view Skill-determination data using the following strategy. First, we sample 3328 single action videos shot by Ego-R, Ego-L, Exo-R and Exo-L cameras and separate them into 1976 training videos and 1352 testing videos. Second, for training videos, we construct video pairs by pairing videos with the same type of action. We do the same for testing videos. Third, given a video pair (v_i, v_j) and their corresponding action quality score s_i and s_j, we regard it as a valid pair if $s_i > s_j + \theta$ is satisfied. Here θ is set as 1.5. By following this strategy, we get 37680 valid pairs (25136 for training and 12544 for testing) for Cross-view Skill Determination. **(2) Baseline model.** We use the state-of-the-art skill determination model RAAN [8] as our baseline model. **(3) Experiment settings.** Following previous works [7, 8], we train an individual model for each task. We sample 500 frames from the videos using the image feature extracted by CLIP [18] as the input of RAAN. For those videos with less than 500 frames, we adopt zero paddings behind the CLIP features and carefully modify the attention module of RAAN to adapt to the masked input.

Experiment. The benchmark results of Cross-view Skill Determination are shown in Tab. A.7. We have similar findings as in Cross-view Sequence Verification benchmark, *i.e.*, training models with all training pairs will not benefit performance on Ego-Ego pairs.

A4 More Comparisons with Related Datasets

A4.1 EgoExo-Fitness v.s. Ego4D

For a fair comparison, in the main paper, we compare EgoExo-Fitness with a subset of Ego4D [9], which contains scenarios of technical full-body actions. All selected scenarios are listed below: {*Dancing, Working out at home, Basketball, Climbing, Outdoor technical climbing/belaying/rappelling (includes ropework),*

Table A.8: More comparison betweens the proposed EgoExo-Fitness and the concurrent Ego-Exo4D [10] dataset. Compared with Ego-Exo4D, the proposed EgoExo-Fitness collects videos of a new scenario (*i.e.*, fitness) and augments data with novel annotations of interpretable action judgment (*i.e.*, text guidance and technical keypoint verification are not provided in Ego-Exo4D).

Datasets	Scenarios	Step	Text guidance	Keypoint verification	Comment	Score	Duration
Ego-Exo4D v1 [10]	Cooking	✓			✓	✓	654h
	Health	✓			✓	✓	124h
	Bike Repair	✓			✓	✓	83h
	Music				✓	✓	216h
	Basketball				✓	✓	61h
	Climbing				✓	✓	88h
	Soccer				✓	✓	96h
	Dancing				✓	✓	99h
Ego-Exo4D v2 [10]	Cooking	✓			✓	✓	564h
	Health	✓			✓	✓	114h
	Bike Repair	✓			✓	✓	82h
	Music				✓	✓	180h
	Basketball				✓	✓	78h
	Climbing				✓	✓	93h
	Soccer				✓	✓	66h
	Dancing				✓	✓	106h
EgoExo-Fitness(Ours)	Fitness	✓	✓	✓	✓	✓	32h

Swimming in a pool/ocean, Football, Going to the gym: exercise machine-class-weights, Yoga practice, Working out outside, Rowing, Skateboard/scooter, Baseball, Roller skating, Playing badminton, Table Tennis, Bowling}.

From Tab. 2 in the main paper, we find that the subset only contains a tiny fraction (about 172h) of videos in the whole Ego4D, which suggests that the egocentric full-body action understanding is rarely addressed even for the largest egocentric video datasets. Compared with Ego4D, EgoExo-Fitness contains synchronized ego-exo videos and novel annotations on how well a fitness action is performed (*i.e.*, annotations of interpretable action judgment), which provides novel resources for future works on view characteristics, multi-view modeling, and action judgment for the egocentric vision community.

A4.2 EgoExo-Fitness v.s. Ego-Exo4D

As supplements to Tab. 3, we provide more comparisons between our datasets and Ego-Exo4D [10] in Tab. A.9. Besides, beyond the similarities and differences discussed in Sec. 3.5, our dataset has a comparative scale with each scenario of *full-body (physical)* actions in Ego-Exo4D (see the Tab. A.9). Note that for fair comparisons, single actions recorded by RGB cameras are considered.

We hope the proposed EgoExo-Fitness can be another resource for studying egocentric full-body action understanding and skill guiding.

Table A.9: Comparisons between Ego-Exo4D [10] on dataset scale. Our dataset has a comparative scale with each scenario of *full-body (physical)* actions in Ego-Exo4D

Datasets	Ego-Exo4D v2 [10]				Ours
Scenarios	Basketball	Climbing	Soccer	Dancing	Fitness
# Tasks/Action Types	3	11	3	2	12
# Single Actions(RGB)	4550	7191	1567	4367	6131

Table A.10: More comparisons with existing full-body action datasets. EgoExo-Fitness has a comparative scale with existing related datasets.

Datasets	MTL-AQA [16]	FineGym [19]	FineDiving [23]	FLAG3D(real) [21]	1st-basketball [1]	WEAR [3]	Ours
# Videos	1412	303	3000	7200	48	18	1276
# Single Actions	1412	32697	3000	7200	-	615	6131

A4.3 EgoExo-Fitness v.s. other related datasets

We also provide more comparisons with existing datasets in Tab. A.10 as supplements to Tab. 2, which show that our dataset has a comparative scale with existing full-body action datasets.

References

1. Bertasius, G., Soo Park, H., Yu, S.X., Shi, J.: Am i a baller? basketball performance assessment from first-person videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2177–2185 (2017)
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: Proceedings of the International Conference on Machine Learning (2021)
3. Bock, M., Moeller, M., Van Laerhoven, K., Kuehne, H.: Wear: A multimodal dataset for wearable and egocentric video activity recognition. arXiv preprint arXiv:2304.05088 (2023)
4. Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., Zisserman, A.: A short note about kinetics-600. arXiv preprint arXiv:1808.01340 (2018)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
6. Dong, S., Hu, H., Lian, D., Luo, W., Qian, Y., Gao, S.: Weakly supervised video representation learning with unaligned text for sequential videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2437–2447 (2023)
7. Doughty, H., Damen, D., Mayol-Cuevas, W.: Who's better? who's best? pairwise deep ranking for skill determination. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6057–6066 (2018)
8. Doughty, H., Mayol-Cuevas, W., Damen, D.: The pros and cons: Rank-aware temporal attention for skill determination in long videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7862–7871 (2019)

9. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18995–19012 (2022)
10. Grauman, K., Westbury, A., Torresani, L., Kitani, K., Malik, J., Afouras, T., Ashutosh, K., Baiyya, V., Bansal, S., Boote, B., et al.: Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19383–19400 (2024)
11. He, T., Liu, H., Li, Y., Ma, X., Zhong, C., Zhang, Y., Lin, W.: Collaborative weakly supervised video correlation learning for procedure-aware instructional video analysis. arXiv preprint arXiv:2312.11024 (2023)
12. Idrees, H., Zamir, A.R., Jiang, Y.G., Gorban, A., Laptev, I., Sukthankar, R., Shah, M.: The thumos challenge on action recognition for videos "in the wild". Computer Vision and Image Understanding **155**, 1–23 (2017)
13. Lin, K.Q., Wang, J., Soldan, M., Wray, M., Yan, R., XU, E.Z., Gao, D., Tu, R.C., Zhao, W., Kong, W., et al.: Egocentric video-language pretraining. Advances in Neural Information Processing Systems **35**, 7575–7586 (2022)
14. Lin, T., Liu, X., Li, X., Ding, E., Wen, S.: Bmn: Boundary-matching network for temporal action proposal generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3889–3898 (2019)
15. Liu, X., Wang, Q., Hu, Y., Tang, X., Zhang, S., Bai, S., Bai, X.: End-to-end temporal action detection with transformer. IEEE Transactions on Image Processing **31**, 5427–5441 (2022)
16. Parmar, P., Morris, B.T.: What and how well you performed? a multitask learning approach to action quality assessment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 304–313 (2019)
17. Qian, Y., Luo, W., Lian, D., Tang, X., Zhao, P., Gao, S.: Svip: Sequence verification for procedures in videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19890–19902 (2022)
18. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of the International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)
19. 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 (2020)
20. Tang, Y., Ding, D., Rao, Y., Zheng, Y., Zhang, D., Zhao, L., Lu, J., Zhou, J.: Coin: A large-scale dataset for comprehensive instructional video analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1207–1216 (2019)
21. Tang, Y., Liu, J., Liu, A., Yang, B., Dai, W., Rao, Y., Lu, J., Zhou, J., Li, X.: Flag3d: A 3d fitness activity dataset with language instruction. In: CVPR (2023)
22. Wang, B., Zhao, Y., Yang, L., Long, T., Li, X.: Temporal action localization in the deep learning era: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
23. Xu, J., Rao, Y., Yu, X., Chen, G., Zhou, J., Lu, J.: Finediving: A fine-grained dataset for procedure-aware action quality assessment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2949–2958 (2022)

24. Zhang, C.L., Wu, J., Li, Y.: Actionformer: Localizing moments of actions with transformers. In: European Conference on Computer Vision (2022)