CoachMe: Decoding Sport Elements with a Reference-Based Coaching Instruction Generation Model

Wei-Hsin Yeh[1,3], Yu-An Su[1], Chih-Ning Chen[1], Yi-Hsueh Lin[1,2], Calvin Ku[2], Wen-Hsin Chiu[2], Min-Chun Hu[2], Lun-Wei Ku [1],
[1]Institute of Information Science, Academia Sinica,
[2]National Tsing Hua University, [3]National Taiwan University
{weihsinyeh168, allen0512911}@gmail.com,
{yuansu, andrewman71, lwku}@iis.sinica.edu.tw,
calvinku@gapp.nthu.edu.tw, whchiu@mx.nthu.edu.tw, anitahu@cs.nthu.edu.tw

Abstract

Motion instruction is a crucial task that helps athletes refine their technique by analyzing movements and providing corrective guidance. Although recent advances in multimodal models have improved motion understanding, generating precise and sport-specific instruction remains challenging due to the highly domain-specific nature of sports and the need for informative guidance. We propose CoachMe, a reference-based model that analyzes the differences between a learner's motion and a reference under temporal and physical aspects. This approach enables both domain-knowledge learning and the acquisition of a coach-like thinking process that identifies movement errors effectively and provides feedback to explain how to improve. In this paper, we illustrate how CoachMe adapts well to specific sports such as skating and boxing by learning from general movements and then leveraging limited data. Experiments show that CoachMe provides high-quality instructions instead of directions merely in the tone of a coach but without critical information. CoachMe outperforms GPT-4o by 31.6% in G-Eval on figure skating and by 58.3% on boxing. Analysis further confirms that it elaborates on errors and their corresponding improvement methods in the generated instructions. You can find CoachMe here: https://motionxperts.github.io/

1 Introduction

Given their strong ability to connect vision with language, recent multimodal models for motion-related tasks have shown significant progress. Existing efforts primarily focus on motion caption (Goutsu and Inamura, 2021; Zhang et al., 2023b) or universal models that perform any motion-related tasks (Guo et al., 2022b; Jiang et al., 2024; Li et al., 2024). These models, trained on large, high-quality datasets such as HumanML3D (Guo et al., 2022a) and KIT-ML (Plappert et al., 2016), excel at understanding motion and generating descriptions such as "A man lifts his left knee to his right elbow." This capability is important for tasks that require subtle movement analysis such as motion tracking (Karaev et al., 2024), physics-based character control (Maloisel et al., 2023) in animation, and robotics. Sports is another area where it is essential to recognize how specific gestures affect overall motion. Excellence in performance relies on movements executed with high precision and outstanding coordination, encompassing both temporal and positional accuracy. Although expert coaches can provide the effective instructions that athletes covet, they are not always readily available.

One solution in scenarios where coaching resources are limited is automatic generation of precise motion instructions. However, there are two challenges in this task: The first is the dynamics of sports, where each discipline has unique movement patterns. Experts gain domain knowledge through years of studying professional techniques (Liu et al., 2024; Chen et al., 2023a). To replicate this expertise, a model must analyze poses from multiple perspectives across different sports, such as changing joint angles, orientations, and temporal variations within a single motion. Note that a motion's meaning differs across sports: in skating, the coordination of knee and shoulder ensures balance and jump execution, whereas in boxing, force transfer from foot to fist determines punching power and strategy.

The second challenge is providing highly informative instructions. Coaches leverage years of experience to guide movement adjustments at precise moments. To achieve similar effectiveness, a model must analyze both physical and temporal perspectives and provide sport-specific, actionable feedback. This includes identifying incorrect body parts, the degree of adjustment, and refining joint alignment or time error, as shown in Fig. 1.

In response to these challenges, we propose

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 29126–29151

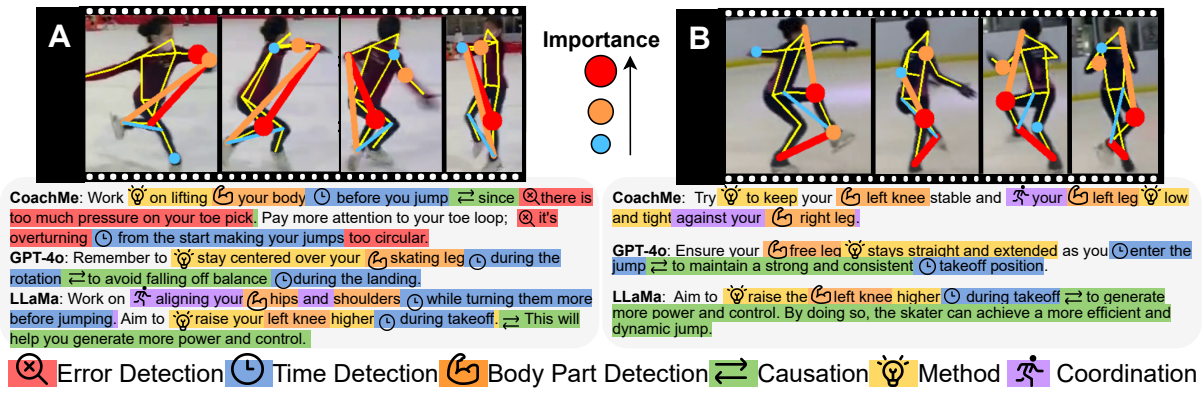

Figure 1: Each video is represented as a sequence of temporally ordered images, with a visualized attention graph of Human Pose Perception (see D.2) overlay. Accompanying each sequence are instructions generated by three models—CoachMe, LLaMa, and GPT-4o—annotated with pictograms that highlight evaluation indicators assessing sport utility and semantic relevance.

CoachMe, a model that obtains domain-specific knowledge from limited data by comparing a learner's motion to a reference motion. By analyzing motion differences and leveraging its intrinsic understanding of movement, CoachMe identifies opportunities for improvement. Furthermore, Basic CoachMe, the base model, facilitates adaptation to sports such as skating and boxing. Moreover, CoachMe's workflow emulates the structured reasoning of a professional coach while instructing athletes, and hence addresses the second challenge. This workflow enables CoachMe to integrate both temporal and physical information, allowing it to generate instructions that not only identify the timing of mistakes but also explain how to improve.

We visualize the weights learned by the model between joints and moving directions and generate instructions to determine whether CoachMe understands the issues in executing the current movement. Additionally, we conduct quantitative and qualitative experiments as well as human evaluation of the performance of CoachMe. Our contributions are threefold: (1) We propose CoachMe, the first reference based model that learns motion differences and generates instructions automatically to offer great precision and sport utility; (2) We conduct extensive experiments to validate the generated instructions by comparing CoachMe to state-of-the-art vision language models, including human evaluation by experts, which is typically challenging to obtain; (3) We construct datasets for instruction generation on two sports—figure skating and boxing—including videos, instructions, and labeled error segments by professional coaches. Datasets will be available upon acceptance.

2 Related Work

Vision language models (Liu et al., 2023a; Zhang et al., 2023c; KunChang Li and Qiao, 2023; Zhang et al., 2023a; Lin et al., 2023) have demonstrated strong capabilities in video-to-text tasks as well as their inverse: text-to-video generation. However, these approaches generally overlook the unique characteristics of pose features and fail to consider the specific demands of different sports. For instance, in skating, certain aspects require special attention, yet existing models tend to generate instructions that are overly general and broadly applicable rather than tailored to the nuances of individual activities.

Beyond general video analysis, human motion modeling has gained traction as a crucial aspect of video understanding. Studies on text-to-motion (Petrovich et al., 2023; Athanasiou et al., 2022; Chen et al., 2023b) generate 3D motion from language descriptions, whereas motion-to-text approaches (Zhang et al., 2023b; Li et al., 2024) describe motion in natural language. For instance, TM2T (Guo et al., 2022b) and MotionGPT (Jiang et al., 2024; Wang et al., 2024) use vector quantized-variational autoencoders (van den Oord et al., 2017) to map motion sequences to discrete representations for motion-related tasks. However, these models focus on general motion descriptions rather than detailed coaching instructions. Liu et al. (2022) and Tanaka et al. (2023) refine specialized skills but are limited to specific sports and rely primarily on visual feedback techniques rather than textual feedback. CoachMe overcomes these limitations by integrating textual instructions with visual explanations, as shown in

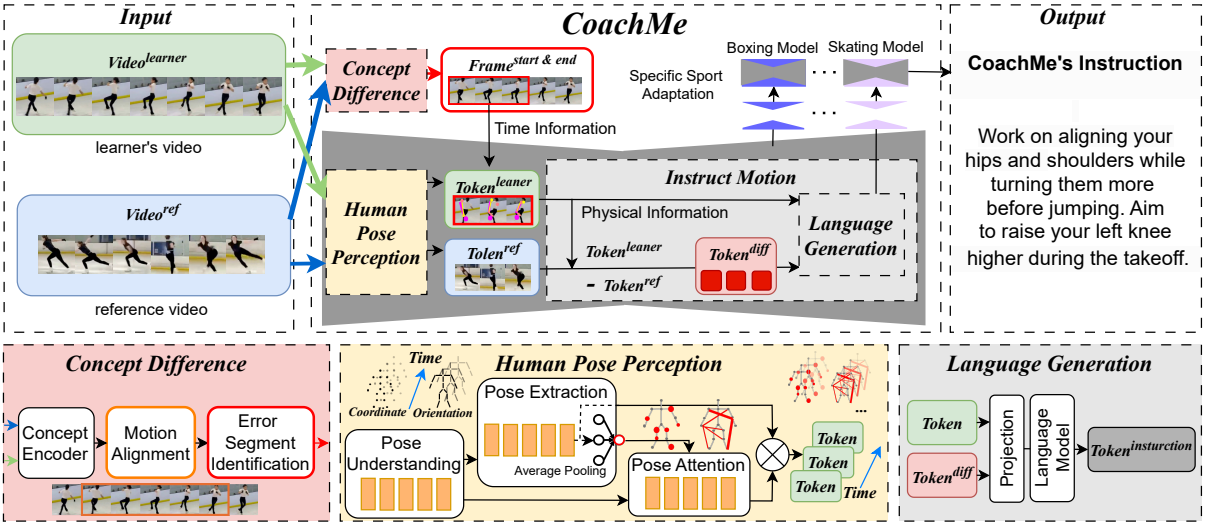

Figure 2: CoachMe architecture comprises three modules: Concept Difference (Sec. 3.1), Human Pose Perception (Sec. 3.2), and Instruct Motion (Sec. 3.3). Instruct Motion compares the motion $Token^{learner}$ with $Token^{ref}$ to obtain the difference $Token^{diff}$ and take $Token^{learner}$ and $Token^{diff}$ as input to the LM to generate instructions.

Fig. 1, while also developing a workflow that is adaptable to different sports.

The idea of concepts is often used for challenging tasks. Concept-based techniques apply single concepts to analyze humor or identify patterns through activation functions (Tennenholtz et al., 2024; Kim et al., 2018) to achieve promising results. In this paper, we extend this to two concepts—*motion* and *difference*—in the sport technology domain. CoachMe aligns videos (Dwibedi et al., 2019; Chen et al., 2022; Kwon et al., 2022) to facilitate synchronization. During this process, motion concept embeddings and difference concept embeddings are generated for further analysis of movement discrepancies.

3 Methodology

To construct the thought process of a coach guiding an athlete, we propose generating instructions based on the differences between the input video and the reference video. CoachMe consists of three modules: Concept Difference, which generates the difference concept; Human Pose Perception, which extracts the motion concept; and Instruct Motion, which processes and integrates the two concepts, as shown in Fig. 2.

3.1 Concept Difference Module

Concept Encoder We define a *concept* as a quantification of performed action, regardless of their viewpoint. A *concept difference* is the deviation between two performances. The concept difference between two frames is computed as

$$c = \mathrm{F}(x_r) - \mathrm{F}(x_l), \tag{1}$$

where x_l represents a frame from the learner's clip, x_r is the corresponding reference frame, and F is a Concept Encoder, which we adopted CARL (Chen et al., 2022). Concept Encoder was trained independently and was frozen during instruction generation training.

Motion Alignment Motion Alignment identifies the interval in the learner video V^L that best corresponds to the reference video V^R and the Concept Difference embedding C is then obtained by subtracting the aligned segment from the reference segment, as shown in Algorithm 1.

Error Segment Identification This module identifies error-prone segments within the target motion. Since frames with higher concept differences are more likely to need correction, we train a model using Concept Difference embeddings C to predict error segments. Given the temporal dependencies in motion sequences, we implement the error segment selection model using a transformer encoder, which captures frame-wise features and temporal relationships. The model outputs the interval requiring instruction as

$$R'_i = R_{i_{SOE:EOE}}, \tag{2}$$

where SOE and EOE denote the start and end of the identified error segment.

Algorithm 1 Motion Alignment

procedure $DTW(A,B,d_a,d_b)$
  $distance, j \leftarrow \infty, -1$
  For $i \in \{0, 1, ..., d_a - d_b\}$ **do:**
    $tmp \leftarrow \text{D}(A_{i:i+d_b}, B_{0:d_b})$ ▷ D:cost matrix
    If $tmp < distance$ **:**
      $distance, j \leftarrow tmp, i$
  return j ▷ Optimal interval's start frame
end procedure
Input : V^L, V^R
$d^L, d^R \leftarrow length(V^L), length(V^R)$
If $d^L < d^R$ **:**
  $V^L, V^R \leftarrow V^R, V^L$
  $d^L, d^R \leftarrow length(V^L), length(V^R)$
$C^L, C^R \leftarrow \text{F}(V^L), \text{F}(V^R)$ ▷ F:Concept Encoder
$j \leftarrow DTW(C^L, C^R, d^L, d^R)$
$C^{L'} \leftarrow C^L_{j:j+d_R}$
$c_t \leftarrow C^R_t - C^{L'}_t, t \in \{0, 1...d_R\}$
$C \leftarrow c_t, t \in \{0, 1...d_R\}$
return C ▷ Concept Difference

3.2 Human Pose Perception Module

Human Pose Perception module contains 3 submodules: Pose Understanding PU, Pose Extraction PE, and Pose Attention PA. Each submodule is based on graph convolutional networks. We first clip V^L according to R'_i, which is the temporal information from Concept Difference. We employ HybrIK (Li et al., 2022) predicts 22 joint coordinates J. We chose HybrIK as it gives better performance when used as a pose estimator of CoachMe. Its inference time is also acceptable since users receive results asynchronously. Details will be discussed (Sec. B). Next, subtracting J outside from J inside results in the joint orientations, O:

$$O_{a,b} = J_a - J_b \quad a, b \in \{0, 1, \ldots, 21\} \tag{3}$$

To capture the mutual influence of J and O, PU shares weights during training. Additionally, to interpret the physical information of motion, PU is trained on G_S, which represents the graph layout of the human skeleton. G_S is constructed from adjacent joint pairs and their distances. Inspired by STA-GCN (Shiraki et al., 2020), this approach incorporates both the temporal dynamics of the motion sequence and the spatial relationships within the skeletal structure. Consequently, PU learns representations of J and O and transforms them into the motion token T. PE is also trained on G_S and generates the motion token T', which mainly

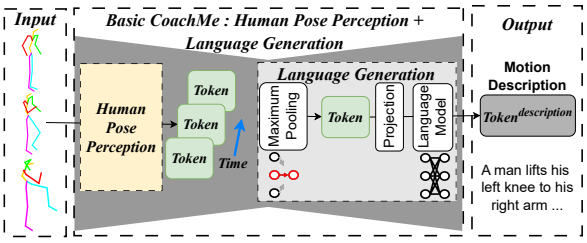

Figure 3: Basic CoachMe, which consists of the Human Pose Perception and Language Generation modules, performs tasks related to motion description.

considers local body parts:

$$T' = \text{PE}_{G_S}(T);\ T = \text{PU}_{G_S}(O \oplus J), \tag{4}$$

where $\oplus$ denotes concatenation. PE applies average pooling on T' to generate the attention joint J_A and attention graph G_A which contain latent relationships between the originally disconnected joints and identifies key joints and relationships. The visualized attention graph, which consists of J_A and G_A, is provided in Figure 1.

$$G_A, J_A = \text{Pool}_{avg}(T') \tag{5}$$

In PA, we use J_A as the input for PA_{G_A} and train it solely with G_A, ensuring a focus on key relationships extracted from G_A. Unlike STA-GCN, CoachMe trains PA solely on G_A. Since PE has already been trained with G_S, it sufficiently captures the physical information within local body parts. This allow PA to focus on learning the physical dynamics of the global body structure. With J_A and G_A, PA_{G_A} propagates the essential attributes of key joints across all previously disconnected joints and local body parts:

$$T'' = \text{PA}_{G_A}(J_A \cdot T), \tag{6}$$

where $\cdot$ denotes the dot product. We utilize both local motion tokens T' and global motion tokens T'' as the final motion tokens, incorporating local, global, spatial, and temporal information:

$$Token = T' \oplus T''. \tag{7}$$

Consequently, Human Pose Perception captures precise motion actions and analyzes subtle motion variations which influence body coordination and are helpful in detecting poor posture.

3.3 Instruct Motion Module

Basic CoachMe was initially pretrained on the HumanML3D dataset to understand the motion token

$Token$ and to learn how to generate textual descriptions of motion, as shown in Fig. 3. To enable CoachMe to provide motion-specific instructions, we integrate temporal segment information from Concept Difference and physical information from Human Pose Perception. We thus obtain the concept motion token $Token$, which represents motion information within the error interval. Next, we compute the concept difference token $Token^{diff}$ by subtracting the learner's motion token from the reference's motion token.

$Token$ and $Token^{diff}$ first undergo maximum pooling along the temporal dimension,which captures each motion token at the most critical moment of the motion sequence that requires further refinement. Next, they are projected into the same latent space. In Projection, understanding the relationship between motion and differences helps identify which aspects of motion need improvement. To enhance adaptability, when fine-tuning the pretrained weights, Basic CoachMe, for the motion instruction task, we apply low-rank adaptation (LoRA) (Hu et al., 2021) to Human Pose Perception and Projection. This aims to refine Human Pose Perception to recognize sport-specific motion patterns and adjust the Projection in Instruct Motion to interpret the relationship of $Token^{diff}$ and $Token$. Finally, the two types of tokens are transformed into instruction I through a language model LM. Because our sports dataset is relatively small (Brigato and Iocchi, 2020), we chose the low-complexity T5 (Raffel et al., 2019), which has only 223M parameters, as our language model LM:

$$I = \mathrm{LM}(\mathrm{Proj}(\mathrm{Pool}_{max}(Token \oplus Token^{diff})), \quad (8)$$

where Proj denotes the projection layer.

4 Experiments

4.1 Datasets

Three datasets were utilized in this paper. **HumanML3D** was adopted for learning general movements, whereas the **Figure Skating** (FS) and **Boxing** (BX) datasets, which we collected, were specifically created for learning sport-specific elements. HumanML3D (Guo et al., 2022a) contains motion sequences from HumanAct12 (Guo et al., 2020) and AMASS (Mahmood et al., 2019). FS comprises 4 types of skating jump videos: single Axel, double Axel, Lutz, and Loop from single learners, annotated by single figure skating coach. Each video is labeled with instructions and corresponding intervals to identify errors, which are denoted as ground truth clips (GT clips), as shown in Table 1. Ground truth clips are employed as the learning targets for the Error Segment Identification module. BX comprises 2 types of boxing technique video: Jab and Cross from multiple learners, annotated by 3 boxing coaches. Each video is labeled with instructions without an error segment. To enhance diversity, we employed GPT-4o to augment the instructions annotated by coaches, as detailed in Section A.3. A summary of these datasets is presented in Table 1. Reference videos were sourced from YouTube coaching content and validated by professional coach.

Dataset	# video train : test	# of motion	GT inst.	Aug.
HumanML3D	23384:4384	-	avg. 3	no
BX	163 : 41	2	3	3
FS	177 : 40	4	1	0
FS(GT clips)	449 : 64	4	1	5

Table 1: Datasets used in experiments. The number of training and test samples are presented for each dataset. FS(GT clips) denotes as ground truth clips that error segments annotated by coach. GT inst. denotes as the number of ground truth instructions. Aug. denoted as the number of augmented instructions for one video.

4.2 Settings

We pretrained Basic CoachMe on HumanML3D to accomplish the motion description task and then adapt it to the FS and BX datasets for motion instruction. To ensure a consistent coordinate system across 3 datasets, we set the pelvis joint as the origin and adopt a local coordinate system. To assess whether referencing also aids descriptions, we experimented with 2 strategies on motion description task: (1) using the first frame of each motion sequence as its reference. (2) employing zero padding to allow the adapted model to learn reference information independently.

We further investigate how the $Token^{diff}$, a key component for contrasting correct and incorrect performances, behaves across modalities. This analysis is motivated by the design of CoachMe, which integrates two distinct modalities: RGB for the Concept Difference and skeleton-based features for the Human Pose Perception. We define 2 settings: CoachMe, which computes $Token^{diff}$ from skeleton-based $Token$ via Human Pose Perception, and CoachMe (RGB), which computes $Token^{diff}$ from RGB-based $Token$ via the Concept Encoder.

We compared CoachMe with GPT-4o (Hurst et al., 2024), LLaMa 3.2 (Dubey et al., 2024), ViLa (Lin et al., 2024) and MiniCPM (Hu et al., 2024).

4.3 Experimental Design

Experiments on description and instruction generation were conducted to evaluate the model's capability in understanding general movements and providing coaching for specific sports. Two experiments were conducted for description generation: (1) Comparing CoachMe to SOTA models for motion description generation. (2) Assessing using references versus not using them when generating descriptions. For the instruction generation task, we conducted four experiments: (1) Comparing CoachMe to SOTA models for motion instruction generation. (2) Studying the gain from applying concept difference. (3) Evaluating how error segment identification (Sec. 3.1) helps in instruction generation. (4) Investigating the impact of different modalities of $Token^{diff}$ on model performance.

4.4 Indicators for Sport Utility Evaluation

To assess the sport utility of the generated instructions, we introduce a set of evaluation indicators that can be applied to various sports. Based on the findings derived from studies on professional athletes (Hülsmann et al., 2017; Bacic and Hume, 2017), effective instruction must fulfill two key aspects, each comprising three essential indicators: The first aspect is that the instruction must **clearly define the problem**, ensuring that athletes understand the issue in their movement. This includes: (1) Detecting **errors** in the motion; (2) Identifying **timing** information; (3) Recognizing **body part** movements. The second aspect is that the instruction must **provide a solution**, offering actionable guidance to help athletes correct their movements. This includes:(4) Identifying **causal relationships** in the sport; (5) Explaining **how to improve** the sport; (6) Describing how body parts **coordinate or interact**. A detailed description of these six indicators will be provided in Section E.2.

5 Results

5.1 Description Generation

Table 2 compares the generated descriptions of movements from SOTA methods. Models are all grounded in the same 3D human motions. Basic CoachMe outperforms others in generating movement descriptions. Human Pose Perception captures 3D motions by treating global and local information equally while considering orientation and coordinates. It thus produces more accurate motion representations than STA-GCN. Then, Instruct Motion transforms these motion tokens into motion descriptions. Interestingly, Table 2 shows that motion description performance is not improved by references, suggesting that understanding general movements does not require predefined standards.

Method	Ref	B1	B4	RG	**BS**
TM2T (2022)	x	61.7	<u>22.3</u>	**49.2**	37.8
MotionGPT (2023)	x	48.2	12.47	37.4	32.4
MotionGPT-2 (2024)	x	48.7	13.8	37.6	32.6
STAGCN*	x	<u>62.8</u>	22.1	47.0	<u>43.5</u>
Basic CoachMe	x	**65.4**	**24.3**	<u>48.6</u>	**45.1**
	v	62.5	20.8	43.3	36.8
	Pad0	59.0	18.2	41.8	35.6

Table 2: Comparison of motion description on HumanML3D. STAGCN* denotes the combination of STAGCN and Instruct Motion. Basic CoachMe combines Human Pose Perception and Instruct Motion. Ref, B1, B4, RG, and BS denote reference, BLEU-1, BLEU-4, ROUGE, and BertScore, respectively.

5.2 Sport Instruction Generation

In this section, we evaluate CoachMe on sport-specific instruction generation, using the FS and BX datasets. To evaluate the role of **reference information**, we compare CoachMe's performance with and without the use of reference video. As shown in Table 3, incorporating reference leads to consistent improvements on both the FS and BX. Human evaluation (see Appendix E.4) supports this finding. This demonstrates the importance of reference-based models in instruction generation.

The experimental results presented in Table 3 indicate that CoachMe consistently achieves higher performance when operating with $Token^{diff}$ based on skeleton modality compared to RGB. The reason is that RGB videos often contain irrelevant elements, such as background distractions, whereas skeleton representations focus solely on motion, leading to more precise instructions for movement refinement. This effect is particularly evident in figure skating, where fast-moving backgrounds introduce additional noise. However, skeleton-based representations may struggle to capture fine-grained joint rotations, such as palm flips.

We analyze the impact of the error segment by providing the model with **(1) ground truth**: instruction intervals labeled by coaches (available only in the FS dataset, where a figure skating coach provided the ground-truth timestamps, while the

Method	Ref	Error segment	BLEU@1	BLEU@4	Rouge	BertScore	G-eval
Figure Skating (FS)							
GPT-4o	x	-	16.2	1.0	14.8	7.5	1.39
LLaMa 3.2	x	-	12.6	0.0	11.4	-4.9	1.31
MiniCpm	x	-	9.7	0.0	11.4	7.2	1.37
ViLa	x	-	8.8	1.8	11.6	11.8	1.27
Basic CoachMe	x	Ground truth	15.0	2.5	16.9	11.0	1.53
CoachMe	v	Ground truth	**24.7**	2.3	16.9	**26.5**	1.73
	v	Aligned Segment	22.2	2.3	**20.0**	11.7	**1.83**
	v	Error Segment	20.8	2.1	15.2	20.5	1.55
CoachMe (RGB)	v	Ground truth	16.9	1.6	15.9	12.0	1.37
	v	Aligned Segment	17.2	4.0	15.8	7.1	1.21
	v	Error Segment	19.4	**4.4**	17.0	8.0	1.57
Boxing (BX)							
GPT-4o	x	-	33.3	0.0	10.0	13.6	1.39
LLaMa 3.2	x	-	16.9	0.0	11.5	3.2	1.20
MiniCpm	x	-	9.1	1.5	9.4	9.1	1.89
ViLa	x	-	6.1	0.0	9.8	1.9	1.40
Basic CoachMe	x	Aligned Segment	41.7	9.4	24.1	36.2	1.85
CoachMe	v	Aligned Segment	38.4	12.3	**28.4**	27.2	**2.20**
	v	Error Segment	**44.5**	**13.4**	25.3	**36.9**	1.61
CoachMe (RGB)	v	Aligned Segment	23.3	6.0	18.2	26.8	1.98
	v	Error Segment	13.9	0.0	11.1	16.5	1.44

Table 3: Comparison of motion instruction generation methods on FS and BX. Ref stands for reference. Performance reported across different CoachMe settings and error segment identification approaches: ground truth (coach-labeled intervals), predicted (model-determined intervals), and no identification (entire aligned video). We also evaluated the consistency between the generated instruction and the ground truth using G-Eval (Liu et al., 2023b).

Dataset	dist-1	dist-2	dist-3
Ground Truth Instruction			
FS	0.115	0.405	0.645
BX	0.037	0.115	0.182
CoachMe's Prediction			
FS	0.233	0.519	0.685
BX	0.070	0.136	0.172

Table 4: Diversity (dist-1, dist-2, dist-3) of ground truth instructions in FS and BX, as well as instructions predicted by CoachMe trained on FS and BX, where dist-n denotes the percentage of distinct n-grams (Li et al., 2016). Higher distinct scores indicate greater diversity.

Model	Good (%)	Neutral (%)	Bad (%)
Figure Skating (FS)			
GPT-4o	20.3	50.0	**29.7**
LLaMa 3.2	18.8	45.3	35.9
CoachMe	**26.6**	43.8	**29.7**
Boxing (BX)			
GPT-4o	46.3	17.1	36.6
LLaMa 3.2	51.2	24.4	24.4
CoachMe	**56.0**	22.0	**22.0**

Table 5: Comparison of instruction quality for 3 models rated by the human evaluator on the FS and BX datasets.

boxing coach did not provide such timestamps); **(2) aligned segment**: the whole movement is trimmed by the Motion Alignment module without error segment identification. **(3) error segment**: intervals selected by the Error Segment Identification module; Note that the Error Segment Identification module achieves an accuracy of 76.14%. Take FS as an example, evaluated by comparing correctly predicted frames with those labeled by coaches: the BertScore results for CoachMe indicate that using either the ground truth or predicted error segments produces instructions that align more closely with coach-labeled instructions compared to those without error segment identification. This highlights the importance of identifying the most error-prone segment for generating instruction.

The higher performance observed on BX compared to FS is primarily attributable to the characteristics of the dataset. BX, constructed from 10 beginner boxers, contains many frequently occurring mistakes, resulting in more consistent feedback from the coaches. Additionally, BX dataset involves only two basic boxing technique with its coaching instructions, leading to lower diversity. In contrast, FS dataset involves 4 jumping movements, which yields higher instruction diversity, as shown in Table 4, indicating a greater modeling challenge effectively addressed by CoachMe. Consequently, CoachMe naturally inherits the respective diversity from each dataset, as demonstrated by the distinct scores of the generated instructions.

5.3 Human Evaluation

We work with figure skating coaches and provide professional evaluations. Table 5 demon-

strates the results of human evaluation. CoachMe received the highest percentage of "Good" ratings (26.6%), outperforming GPT-4o (20.3%) and LLaMA (18.8%).It also aligned more closely with key instructional elements, excelling in identifying the correct timing, relevant body parts, causal relationships, and corrective methods—indicators previously defined in Experiments (Sec 4.4). These factors consistently appeared when CoachMe was rated as the best, confirming their positive impact. Detailed breakdowns can be found in Appendix (Sec. E.6). Notably, CoachMe demonstrated the strongest ability to capture coordination-related aspects and analyze complex body mechanics.

As for the performance on the BX dataset, CoachMe received the highest percentage of "Good" ratings (56.0%) and the lowest percentage of "Bad" ratings (22.0%), as shown in Table 5. The superior performance observed on the BX dataset compared to FS is consistent with the findings discussed in Section 5.2. We interviewed the boxing coach who conducted the human evaluation. The coach emphasized that in sports instruction, concise instructions are more effective and easier for learners to follow. Moreover, the boxing videos consist of beginners performing 2 fundamental techniques, making detailed instructions unsuitable. In contrast, LLaMA and GPT-4o tend to generate overly complex feedback (See 6.1. This preference for concise, targeted instructions likely contributed to CoachMe's superior performance on the BX dataset. Details can be found (see Sec. E.7).

6 Discussion

In this section, we evaluate the generated instructions through indicators (Sec. 4.4) that assess both their sport utility and semantic relevance. We compare CoachMe to LLaMa and GPT-4o to show its indispensability. Last, we delve deeper into the styles (Sec. 6.1), sport indicators (Sec. 6.2) and the design of CoachMe (Sec. 6.3).

6.1 Finding 1: CoachMe Generates More Accurate Instruction

We analyze the relation between the 6 indicators and the instruction quality. GPT-as-a-judge helps annotate the indicators mentioned in the generated instructions. We compare CoachMe to LLaMa and GPT-4o. Figures 4 show the percentage of figure skating instructions in which each indicator is mentioned and the corresponding distribution

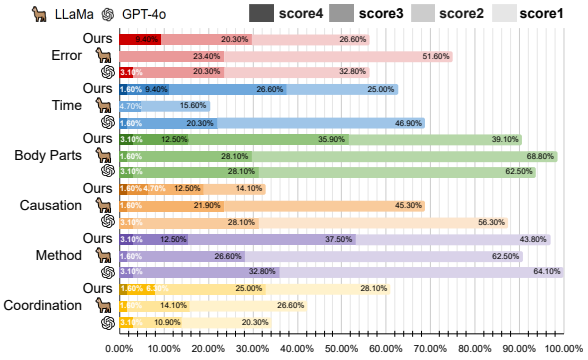

Figure 4: The horizontal bar chart represents the total number of instructions assigned to each indicator (Sec. 4.4), with colors indicating their respective G-eval scores. Higher G-eval scores indicate more informative and effective guidance that aligns more closely with the ground truth. The G-eval scores range from 1 to 5.

of their G-eval scores. Details can be found (See Sec. E.2). Boxing indicator frequencies and G-eval score distributions are analyzed in Section E.7.

The results reveal that although GPT-4o and LLaMa can generate instructions relevant to these indicators (their total bar length is long), they struggle to produce high-scoring instructions (their dark bar length is short). The majority of their guidance is too general or ineffective, resulting in G-eval scores of 1 or 2 (see their longer light colored bar). In other words, LLaMa and GPT-4o are good at playing the role and instructing in a coach's tone but their content is inaccurate, whereas CoachMe consistently generates precise and actionable instructions. Notably, nearly 60.9% of CoachMe's instructions incorporate coordination-related feedback, highlighting its strength in capturing relationships between body parts. This ability is further reflected in its attention mechanism.

As shown in Fig. 1, thicker and redder lines indicate more important movement relationships. CoachMe's attention graph automatically captures joint relations and aligns closely with its instructions. For example, the model highlights the left leg and right leg connection, which leads to the instruction "Keep your left leg low and tight," demonstrating its ability to attend to relevant body parts for precise instruction.

6.2 Finding 2: Good Instructions Identify Problems and Provide Solutions

As shown in Fig. 5, we study the pairwise relationship between indicators and investigate which combinations contribute most to high quality in-

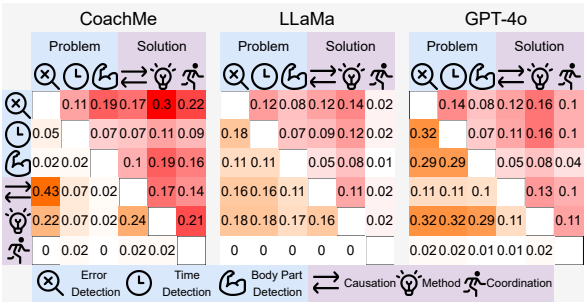

Figure 5: Effectiveness of indicator combinations across models is measured using G-Eval scores, where higher percentages indicate better guidance quality. In the evaluation matrix, red numbers in the upper-right triangle represent scores on the FS dataset, highlighting that CoachMe consistently outperforms GPT-4o and LLaMa. Orange numbers in the lower-left triangle show scores on the BX dataset. Each number denotes the percentage p, calculated based on Eq. 5.

structions. Percentage p in Fig. 5 is calculated as

$$p = \frac{\Sigma_{i=1}^{N} S \text{ of instructions containing two indicators}}{(\text{maximum of } S \times N)}, \tag{9}$$

where N denotes the total number of instructions and S denotes the G-eval score. **Error + Coordination** (0.22) is highly effective, indicating that integrated body coordination feedback ensures athletes adjust their full-body mechanics rather than isolated actions. **Error + Method** achieves the highest contribution (0.3), as identifying errors while providing corrective strategies significantly assists refining movements. For instance, in Fig. 1-A, the issue is effectively pinpointed and clear corrective instruction is suggested. By contrast, **Time + Body Part** (0.07) is less effective as no actionable guidance is given.

Figure 5 illustrates the accuracy of instructions which meet two indicators. Dark color represents a high average G-eval score. With more darker areas, results show that CoachMe generates more accurate instructions. Moreover, CoachMe generates more coordination-related instructions, which confirms that the proposed Human Pose Perception (Sec. 3.2) enables CoachMe to analyze full-body movement. Particularly, CoachMe achieves relatively high scores on all dual indicators, showing that the instructions it generates consider multiple aspects like experts. Overall, providing a precise solution in and of itself helps greatly; further identifying the problem while suggesting a solution only improves the instruction. Findings 1 and 2 suggest the superiority of CoachMe from these aspects.

6.3 Finding 3: CoachMe Captures Sport-Specific Instructional Patterns

CoachMe demonstrates a strong ability to replicate the distribution of sport indicators found in ground-truth instructions annotated by professional coaches. Its predicted instruction distributions in both figure skating (FS) and boxing (BX) closely align with those of the actual datasets (Fig. 6, 7, and 8), indicating effective domain adaptation. In the BX dataset, both instruction predicted by CoachMe and ground truth rarely mention sport indicator "Time Detection" and "Coordination." Expert feedback reveals that beginner-level boxing instruction favors simple, body-part-specific guidance over complex relational cues. In addition, coach also indicates that a basic punch video is very short, temporal errors detection is not useful in such motion. Fortunately, CoachMe learns this pattern well, aligning its outputs with the coaching style used for fundamental actions. These findings underscore the critical role of domain-specific knowledge, given that each sport has its own distinct movement characteristics. This aligns with CoachMe's core design principle—namely, its ability to flexibly adapt to various sports by learning sport-specific patterns through efficient, lightweight adaptation modules.

7 Conclusion

We propose CoachMe, a reference-based motion-to-instruction model that generates tailored instructions for sports. With the injection of concept difference, human pose perception, and instruct motion, CoachMe simulates a coach's thinking process, which identifies deviations from standard motions and focuses on both crucial body parts and their physical movements. Through comprehensive evaluations using quantitative performance metrics, human evaluation, and visual explanations on skating and boxing datasets, we demonstrate CoachMe's ability to provide high accuracy and good sport utility in generated instructions for learners.

CoachMe achieves state-of-the-art results for both motion description and instruction generation. Moreover, it represents a significant step forward to bridge the gap between artificial intelligence and human expertise in athletic training. This lightweight, standalone, high-performance, adaptive model yields good opportunities for deployment in various sport scenarios.

Limitations

CoachMe is primarily designed for beginner- and intermediate-level practitioners. Our dataset consists of practice videos from novice athletes, and the generated instructions focus on fundamental movement correction. Therefore, CoachMe may not be well-suited to provide guidance on advanced techniques or professional-level performance. Future work could explore adapting the model to higher-level athletes by incorporating expert demonstrations and more complex movement evaluation.

In the practical implementation of CoachMe, we prioritize maintaining a consistent coaching style to authentically replicate the experience of working with a real coach. This approach ensures that users receive clear, coherent, and dependable guidance, as demonstrated throughout our paper. During our research, interviews with athletes revealed a clear preference for personalized and consistent coaching styles, as opposed to general or mixed approaches. Nevertheless, the real world offers diverse coaching styles, and each athlete has their own unique preferences. To meet this demand, CoachMe is designed with a flexible architecture that allows the creation of new virtual coaches simply by gathering instructional datasets from real-world coaches. This adaptability not only enables athletes to select virtual coaches aligned with their individual needs and preferences, but also empowers CoachMe's scalability to seamlessly incorporate coaching data from a wide range of experts — including globally recognized professionals. In summary, CoachMe currently learns only a specific teaching style based on our dataset, which limits its ability to represent the diverse speaking styles and instructional approaches of coaches in real-world scenarios. Future work could focus on incorporating a broader range of coaching styles to enhance its generalizability.

One limitation of CoachMe stems from the absence of figure skating and boxing data in the HumanML3D, which can lead to misclassification of actions. To address this, future work could explore integrating an additional action classification module to provide more explicit movement categorization. Furthermore, augmenting HumanML3D with sport-specific data—such as curated figure skating or boxing sequences—could significantly enhance CoachMe's ability to accurately recognize subtle movement distinctions. By enriching the dataset with targeted sports movements and creating sport-specific versions of CoachMe tailored to the unique features of each discipline, we can continuously solve this issue.

Ethics Statement

All participants whose videos are included in our dataset provided informed consent, acknowledging that their videos would be used for research purposes. For participants who are minors, we obtained consent forms from their parents or legal guardians. We ensured that all data collection and usage practices complied with ethical guidelines and respected the privacy and rights of the participants. Additionally, no personally identifiable information was revealed in the dataset, and all files were processed to ensure anonymity.

Acknowledgments

This work is supported by the National Science and Technology Council of Taiwan under grants 114-2425-H-007-002 and 114-2425-H-007-001. We extend our appreciation to Kristina Stepanova, a professional figure skating coach, for her insightful guidance on figure skating techniques, and for also providing expert human evaluation. We also thank Shirley Tzeng for contributing figure skating movement data. These movements and instructional descriptions were used to construct the Figure Skating (FS) dataset. We also gratefully acknowledge the support of National Tsing Hua University, with special thanks to professional boxing coaches Kai-Chun Hong and Chia-Yao Chung for their expert annotations and contributions of instructional data on boxing movements. These movements and instructional descriptions were used to construct the Boxing (BX) dataset. We also thank Chen-Wei Pan for providing professional human evaluation.

References

Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. 2022. Teach: Temporal action compositions for 3d humans. In *International Conference on 3D Vision (3DV)*.

Boris Bacic and Patria Hume. 2017. Computational intelligence for qualitative coaching diagnostics: Automated assessment of tennis swings to improve performance and safety. *CoRR*, abs/1711.09562.

L. Brigato and L. Iocchi. 2020. A close look at deep learning with small data. *Preprint*, arXiv:2003.12843.

Chien-Chang Chen, Chen Chang, Cheng-Shian Lin, Chien-Hua Chen, and I. Cheng Chen. 2023a. Video based basketball shooting prediction and pose suggestion system. *Multimedia Tools Appl.*, 82(18):27551–27570.

Minghao Chen, Fangyun Wei, Chong Li, and Deng Cai. 2022. Frame-wise action representations for long videos via sequence contrastive learning. *Preprint*, arXiv:2203.14957.

Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. 2023b. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18000–18010.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. 2019. Temporal cycle-consistency learning. *Preprint*, arXiv:1904.07846.

Yusuke Goutsu and Tetsunari Inamura. 2021. Linguistic descriptions of human motion with generative adversarial seq2seq learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4281–4287.

Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. 2022a. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161.

Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. 2022b. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. *Preprint*, arXiv:2207.01696.

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. 2024. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*.

Felix Hülsmann, Stefan Kopp, and Mario Botsch. 2017. Automatic error analysis of human motor performance for interactive coaching in virtual reality. *CoRR*, abs/1709.09131.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2024. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36.

Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. 2024. Cotracker3: Simpler and better point tracking by pseudo-labelling real videos. In *Proc. arXiv:2410.11831*.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *Preprint*, arXiv:1711.11279.

Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. 2020. Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yi Wang Yizhuo Li Wenhai Wang Ping Luo Yali Wang Limin Wang KunChang Li, Yinan He and Yu Qiao. 2023. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*.

Taein Kwon, Bugra Tekin, Siyu Tang, and Marc Pollefeys. 2022. Context-aware sequence alignment using 4d skeletal augmentation. *Preprint*, arXiv:2204.12223.

Nayoung Lee, Kartik Sreenivasan, Jason D. Lee, Kangwook Lee, and Dimitris Papailiopoulos. 2023. Teaching arithmetic to small transformers. *Preprint*, arXiv:2307.03381.

Chuqiao Li, Julian Chibane, Yannan He, Naama Pearl, Andreas Geiger, and Gerard Pons-Moll. 2024. Unimotion: Unifying 3d human motion synthesis and understanding. *arXiv preprint arXiv:2409.15904*.

Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. 2022. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. *Preprint*, arXiv:2011.14672.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*,

pages 110–119, San Diego, California. Association for Computational Linguistics.

Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. 2024. Vila: On pretraining for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26689–26699.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. In *NeurIPS*.

Jingyuan Liu, Nazmus Saquib, Zhutian Chen, Rubaiat Habib Kazi, Li-Yi Wei, Hongbo Fu, and Chiew-Lan Tai. 2022. Posecoach: A customizable analysis and visualization system for video-based running coaching. *IEEE Transactions on Visualization and Computer Graphics*.

Jingyuan Liu, Nazmus Saquib, Chen Zhutian, Rubaiat Habib Kazi, Li-Yi Wei, Hongbo Fu, and Chiew-Lan Tai. 2024. Posecoach: A customizable analysis and visualization system for video-based running coaching. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):3180–3195.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. G-eval: Nlg evaluation using gpt-4 with better human alignment. *Preprint*, arXiv:2303.16634.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. 2015. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. 2019. Amass: Archive of motion capture as surface shapes. *Preprint*, arXiv:1904.03278.

Guirec Maloisel, Christian Schumacher, Espen Knoop, Ruben Grandia, and Moritz Bächer. 2023. Optimal design of robotic character kinematics. *ACM Trans. Graph.*, 42(6).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Mathis Petrovich, Michael J. Black, and Gül Varol. 2023. TMR: Text-to-motion retrieval using contrastive 3D human motion synthesis. In *International Conference on Computer Vision (ICCV)*.

Matthias Plappert, Christian Mandery, and Tamim Asfour. 2016. The kit motion-language dataset. *Big data*, 4(4):236–252.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Katsutoshi Shiraki, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. 2020. Spatial temporal attention graph convolutional networks with mechanics-stream for skeleton-based action recognition. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*.

Ryota Tanaka, Tomohiro Suzuki, Kazuya Takeda, and Keisuke Fujii. 2023. Automatic edge error judgment in figure skating using 3d pose estimation from a monocular camera and imus. *Preprint*, arXiv:2310.17193.

Guy Tennenholtz, Yinlam Chow, Chih-Wei Hsu, Jihwan Jeong, Lior Shani, Azamat Tulepbergenov, Deepak Ramachandran, Martin Mladenov, and Craig Boutilier. 2024. Demystifying embedding spaces using large language models. *Preprint*, arXiv:2310.04475.

Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. Neural discrete representation learning. *CoRR*, abs/1711.00937.

Yuan Wang, Di Huang, Yaqi Zhang, Wanli Ouyang, Jile Jiao, Xuetao Feng, Yan Zhou, Pengfei Wan, Shixiang Tang, and Dan Xu. 2024. Motiongpt-2: A general-purpose motion-language model for motion generation and understanding. *Preprint*, arXiv:2410.21747.

Hang Zhang, Xin Li, and Lidong Bing. 2023a. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023b. T2m-gpt: Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. 2023c. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

A Dataset Details

A.1 Dataset Construction

We divided the dataset into training and testing sets in a $4:1$ ratio. Given that a single figure skating video can be split into multiple video clips corresponding to error segment annotated by professional figure skating coach, we ensured that clips from the same video did not appear in both the training and testing sets. This approach was implemented to prevent data leakage and ensure that the model's performance evaluation is based on unseen data. Therefore, the number of training videos in the Figure Skating (FS) dataset is 292, with 64 videos for testing, as shown in Table 4.1. The HumanML3D dataset has a frame rate of 20 frames per second (FPS), the Boxing (BX) dataset operates at 60 FPS, and the FS dataset is recorded at 30 FPS.

A.2 Data Preprocessing

To best mitigate noise originated from the background, for all rgb settings and videos we track all people in the scene using YOLOv7, manually select the main person index then resize the video into 224x224, centered at the main character.

In human pose perception, we employ HybrIK (Li et al., 2022) to predict 22 joint coordinates in a local coordinate system following the SMPL (Loper et al., 2015) format for videos from the Figure Skating and Boxing datasets. To ensure consistency between the pretraining and finetuning settings, we localize the coordinates in the Human ML3D dataset. Localization involves subtracting the coordinates of the first index joint, which is the pelvis. After localization, the motion videos appear to rotate around the pelvis.

A.3 Data Augmentation Template

In Section 4.1, we incorporate the GPT-4 API to diversify the original dataset. The template we used is shown in Table 6 and Table 7.

Because the figure skating coach provides one or multiple error segments for each figure skating video, along with the corresponding original ground-truth instruction for each segment, we apply data augmentation accordingly. Specifically, for each ground-truth instruction, we generate five augmented instructions using five different templates, as shown in Table 6. As a result, each annotated error segment, a ground-truth clip, has six associated instructions: one original and five augmented versions. For the FS dataset, the augmented instructions maintain a high degree of similarity with their corresponding original instructions. The average similarity score is 0.93, with 100.00% of the instructions scoring above 0.8, and 92.40% above 0.9.

The augmented instructions for the BX dataset show even higher similarity scores with their originals. The average similarity score is 0.95, with 100.00% above 0.8 and 98.38% above 0.9. This is because each boxing video includes three ground-truth instructions annotated independently by three different coaches. When applying augmentation to these three original instructions, we use the same template as shown in Table 7. As a result, each raw video (without coach-annotated error segmentation) yields six labels in total: three original and three augmented instructions.

Additionally, we appended a restrictive prompt—"... not begin with a directive verb..."—to the end of each prompt template. This decision stems from our observation that when models are asked to rephrase an instruction—which typically emphasizes keeping the tone positive, encouraging, and supportive—they often adopt an overly encouraging tone and frequently begin with directive verbs such as "Keep...". To counter this tendency, we explicitly prohibit such language and instead require a neutral tone. Furthermore, to minimize hallucination during data augmentation using GPT-4o, we also include the constraint: "Do not introduce any information that is not present in the target instruction." This is designed to ensure that CoachMe learns from augmented data without incorporating hallucinated or fabricated content.

A.4 Human Annotation

We employed human annotators for data labeling, with skating annotations provided by professional coaches from Europe and Asia, and boxing annotations conducted by members of a college boxing team. Annotators were informed that their labeled data would be used to train models. Annotators were recruited based on their domain expertise and compensated at a rate of $50 per hour. The data collection protocol was reviewed and approved by the Institutional Review Board (IRB).

B Pose Estimator

We selected HybrIK (Li et al., 2022) for pose estimation in this study because it represents the cur-

Role	Content
system	You are an experienced figure skating coach who specializes in helping students improve their skating skills, particularly with the {motion type} jump. Your task is to rephrase the instruction. Please follow this guideline when rewriting: Guideline: 1. Use simple and clear language that beginners can easily understand and apply. 2. Maintain a clear and neutral tone with a professional and objective style. 3. Feel free to omit parts of the original instruction that are not particularly helpful. 4. Avoid phrasing that sounds too strict or overly commanding. 5. Focus on offering constructive suggestions that help students feel motivated to improve. If the target instruction does not begin with a directive verb such as "Keep", "Try", "Focus on", "Ensure", "Consider", "Aim", "Avoid", "Make sure", or "Remember", you should avoid introducing one in the rephrased version. Maintain a neutral tone. Do not introduce any information that is not present in the target instruction. Please provide exactly one alternative way to rephrase the instruction.
user	Target instruction: {instruction}

Table 6: Data augmentation template for GPT-4, where {motion type} refers to the current jump type (Axel, Lutz, Loop) and {instruction} refers to the original annotation.

Role	Content
system	You are an experienced boxing coach who specializes in helping students improve their boxing skills, particularly with the {motion type} technique. Your task is to rephrase the instruction. Please follow this guideline when rewriting: Use simple and clear language that beginners can easily understand and apply. Guideline: 1. Use simple and clear language that beginners can easily understand and apply. If the target instruction does not begin with a directive verb such as "Keep", "Try", "Focus on", "Ensure", "Consider", "Aim", "Avoid", "Make sure", or "Remember", you should avoid introducing one in the rephrased version. Maintain a neutral tone. Do not introduce any information that is not present in the target instruction. Please provide exactly one alternative way to rephrase the instruction.
user	Target instruction: {instruction}

Table 7: Data augmentation template for GPT-4, where {motion type} refers to the current boxing technique (Jab and Cross) and {instruction} refers to the original annotation.

Pose Estimator	Figure Skating (FS)	Boxing (BX)
HyBriK	21sec	35sec
VIBE	10sec	14.8sec

Table 8: Inference Time Comparision of different pose estimator on Figure Skating and Boxing datasets.

rent state-of-the-art approach. For comparison, we also conducted experiments using VIBE (Kocabas et al., 2020). The Mean Per Joint Position Error (MPJPE) scores for both methods are presented in Table 8, and their respective inference times are provided in Table 9. From these tables, we observe that the difference in inference times between HybrIK and VIBE is relatively minor in our use case, where users upload videos and await generated instructions. Thus, our system offers flexibility: users who prioritize faster feedback can opt for a lightweight model like VIBE, while those who prefer enhanced accuracy only need to wait an additional ten seconds to benefit from HybrIK's superior precision.

Furthermore, to evaluate the temporal smoothness of the features extracted by HyBriK and VIBE, we computed the differences per joint frame to frame in the FS and BX datasets, as shown in Table 10 below. The metric is defined as:

$$\text{Smoothness} = \frac{1}{T-1}\sum_{t=1}^{T-1} \|J_{t+1} - J_t\|_2 , \quad (10)$$

where J_t denotes the 3D joint positions at frame t, and T is the total number of frames. A lower value indicates smoother motion over time. It can be seen that HyBriK produces significantly lower differences than VIBE, indicating smoother pose estimation on our datasets. This effectively reduces joint misalignment and skeleton drift during fast actions, making HyBriK more suitable for consistently generating accurate instructions.

To further examine the impact of pose estimation quality on instructional output, we conducted an experiment comparing HybrIK and VIBE as the pose

Pose Estimator	3DPW	Human3.6	MPI-INF-3DHP
HyBriK	71.6	47.0	91.0
VIBE	82.9	65.6	96.6

Table 9: Comparison of HyBriK and VIBE on 3DPW, Human3.6M, and MPI-INF-3DHP Using MPJPE.

Model	Figure Skating (FS)	Boxing (BX)
HyBriK	0.0260±0.0267	0.0043±0.0030
VIBE	0.1340±0.1604	0.0396±0.0443

Table 10: Per-joint frame-to-frame differences.

	B1	B4	RG	BS	G-eval
Dataset	**Figure Skating (FS)**				
HyBriK	24.7	2.3	16.9	26.5	1.73
VIBE	16.3	2.1	13.1	0.06	1.35
Dataset	**Boxing (BX)**				
HyBriK	43.9	15.3	26.6	39.8	1.63
VIBE	43.1	13.5	26.0	36.3	1.65

Table 11: Comparison of motion instruction generation methods on FS and BX with different pose estimator. We evaluated the consistency between the generated instruction and the ground truth using different NLP metric and G-Eval. B1, B4, RG, and BS denote BLEU-1, BLEU-4, ROUGE, and BertScore, respectively.

Module	FS	BX
Motion Alignment	0.37sec	0.30sec
HyBrIK	21sec	35sec
Human Pose Perception & InstructMotion	1.93sec	2.09sec

Table 12: The inference times of Motion Alignment, HyBrIK, and Human Pose Perception on the Figure Skating (FS) and Boxing (BX) datasets.

Module	# of Parameters
Basic CoachMe	305.54M
Lora of Figure Skating	5.64M
Lora of Boxing	5.64M
Trainable parameters of CoachMe	232.88M
T5-base	223M

Table 13: The number of parameters in Basic CoachMe and the trainable parameters in CoachMe for sport-specific tasks.

estimators for CoachMe. The results are presented in Table 11.

As shown in Table 11, instruction quality declined across metrics (BLEU, ROUGE, BERTScore) when VIBE was used instead of HybrIK. These findings suggest that in complex, high-precision domains such as professional sports, even subtle inaccuracies in pose estimation can lead to misleading or suboptimal feedback.While HybrIK consistently provides more accurate results, the performance gap—particularly in simpler cases like boxing—is relatively moderate. Reflecting this, CoachMe is designed with two pose estimation modes: a fast-response mode using VIBE for immediate feedback, and a high-precision mode using HybrIK for more accurate guidance. In this paper, we report results based on HybrIK to ensure the highest possible instruction quality and consistency across experiments.

While HybrIK introduces a higher computational overhead compared to lighter models, feedback from professional athletes and coaches in real-world scenarios indicates that the overall evaluation time is acceptable. We observed that the processing time did not negatively impact the usability or practicality of the system.

C Model

C.1 Hyperparameter Settings

For the optimizer, we use AdamW with a learning rate of 1e-4 and 5000 warm-up steps for both the pre-training and fine-tuning settings.

For pretraining, we used an A100 GPU, with a total training time of 1720 minutes (17 min 12 sec $\times 100$). The batch size was set to 16, with a maximum of 50 epochs.

For finetuning, we set the learning rate to 1×10^{-4} with a maximum of 200 epochs and 5000 warmup steps. The batch size is set to 4. LoRA configurations for Human Pose Perception and Projection use both bias $=$ none, $r = 32$, alpha $= 64$, and dropout $= 0.1$. The dropout rate is set to 0.5 for Human Pose Perception and 0.1 for the projection layer. Training was carried out on an RTX 4090, boxing training took 500 minutes (2.5×200) and skating training 200 minutes (1×200).

As shown in 14, we can see HybrIK preprocessing accounts for 90% of the inference time, while CoachMe only requires the remaining 10%.

The number of parameters in Basic CoachMe and the trainable parameters in CoachMe for sport-specific tasks.

For Human Pose Perception, we set the number of kernels in the convolutional neural network to 1024. This is to capture diverse motion details in motion descriptions and various error types in motion instructions.

C.2 Model Configuration

This section presents a detailed layer-wise analysis of each model's architecture, elaborating on the structure and functionality of individual components. Table 5 provides a comprehensive overview of the parameter counts along with the corresponding architectural configurations for all modules.

Concept Difference					
Concept Encoder					
Module	ResNet-50 (finetune)	MLP	Transformer Encoder (x3)	Final Projection	
# param.	23.5M	1.7M	3.1M	0.13M	
shape	→2048	→256	→256	→128	
Error Segment Identification					
Module	Input FC	Self-Attention	FFN	Layers Norm	Output FC
# param.	2K	1K	65K	64	34
shape	16->16	->2048	->16	->16	->2
Human Pose Perception					
Pose Understanding					
Block	0	1	2	3	4
# param.	14K	20K	58K	70K	22K
shape	6->32	->32	->32	->64	->128
Pose Extraction					
Block	0	1	2	3	4
# param.	267K	267K	859K	1M	1M
shape	128->128	->128	->256	->256	->256
Layer	BN	Conv	AttnJ	AttnG	
# param.	512	262K	1K	90K	
Pose Attention					
Block	0	1	2	3	4
# param.	172K	172K	645K	636K	636K
shape	128->128	->128	->256	->256	->256
Instruct Motion					
Projection					
Layer	0	1	2	3	
# param.	51K	34K	34K	42K	
shape	512->512	->512	->512	->768	
Language Model : T5-base					
# param.	223M				

Table 14: Layer-wise architectural configurations and parameter counts of all models. # param. is denoted as the number of parameters. BN is denoted as the batch normalize. Conv is denoted as the convolution. AttnJ is denoted as the attention joint. AttnG is denoted as the attention graph.

C.2.1 Concept Difference

The Concept Encoder consists of four main components: a ResNet-50 backbone with partial finetuning, a lightweight MLP transformation module, a 3-layer Transformer Encoder, and a final projection head. The ResNet-50 backbone is finetuned from stage 4 onward, transforming the input RGB clip into a 2048-dimensional feature representation. This is followed by an MLP comprising three fully connected layers that reduce the dimensionality to 256. The Transformer Encoder further processes the temporal sequence of features across three layers, each with multi-head self-attention and feed-forward blocks. The final projection head maps the representation to a 128-dimensional embedding space for motion alignment.

For the Error Segment Identification module, we adopt a simple transformer-style architecture. The input first passes through a linear layer to expand the feature space from 16 to 16 dimensions, followed by a single-layer self-attention mechanism. This is followed by a feed-forward network with an intermediate dimensionality of 2048. Layer normalization is applied before and after the attention and feed-forward layers. A final output layer maps the sequence to a binary prediction indicating whether a frame belongs to an error segment.

C.2.2 Human Pose Perception

In Human Pose Perception, Pose Understanding, Pose Extraction, and Pose Attention all consist of five identical blocks. Each block includes a spatial convolution, implemented as a single 2D convolution layer, followed by a temporal convolution composed of a batch normalization layer, a ReLU activation function, a 2D convolution layer, another batch normalization layer, and a second ReLU activation function, applied sequentially. Additionally, each block incorporates a residual connection that consists of a 2D convolution layer followed by a batch normalization layer.

After the Pose Extraction blocks, the output is first processed by a batch normalization layer followed by a convolution layer to produce the attention embedding. The attention embedding is then further processed sequentially by a convolution layer, a batch normalization layer, and a linear

interpolation operation. Finally, a sigmoid activation function is applied to generate the attention joint. In parallel, the attention embedding first undergoes average pooling, convolution, and batch normalization, then passes through tanh and ReLU activations to generate the attention graph.

In terms of the graph, except for Pose Attention, which utilizes the attention graph, the rest are based on the spatial graph.

C.2.3 Instruct Motion

The Projection applies temporal average pooling, followed by three consecutive blocks, each consisting of a linear layer, a batch normalization layer, and a ReLU activation function applied in sequence. Then, a final linear layer to map from 512 to 768 dimensions for T5-base input.

We choose beam search as our decoding strategies in InstructMotion, as discussed in Section 3.3. As beam search typically generates better descriptions of motion sequences by considering more candidate options, we have chosen it over greedy search. We set the beam size of beam search is 3.

For tokenizer, we utilize the AutoTokenizer from the Hugging Face Transformers library to efficiently process input sequences. This ensures optimized tokenization while maintaining compatibility with the T5-base model. For the T5 configuration, we use T5ForConditionalGeneration from the Hugging Face Transformers library to initialize the model with a predefined configuration. This guarantees consistency in the model architecture and parameter settings, while leveraging the pretrained weights of T5-base for effective sequence-to-sequence generation.

C.3 Pretrain and Finetune

We adopt the idea of Chain of Thought (Lee et al., 2023). We use "Motion Description" during pretrain and "Motion Instruction" during finetune as the start tokens separately. Therefore, model can distinguish whether to generate motion description or instructional language according to the task prompt given from the start tokens.

D Visualization

D.1 t-SNE Visualization of Learner-Standard Pair

In Figure 10 we present more examples of selecting the minimum distance and visualized embeddings.

In illustration **A**, we show that the gray frames are not selected since the frame in the standard video represents preparation for the Axel, which is not included in the input video. Illustrations **B** and **C** showcase the aligned start and end frames by computing the minimum distance using the DTW cost matrix. Each illustration is associated with its corresponding DTW cost matrix, where the red dot denotes the minimum value.

D.2 Visualize Attention of Human Pose Perception

To explain why certain motion elements are generated in the instructions, we visualize the attention graph and joints used in Human Pose Perception. As shown in Figure 1, We visualize the top-3 important relationships of two joints, which is recorded in attention graph, and top-3 important joints, which is recorded in attention joints, that learned by Human Extraction. We set the number of attention graph to 4. Figure 1 shows that the different attention graphs can pay attention on different human body-parts.

E Evaluation

E.1 Metrics

Following Guo et al. (2022b) and Jiang et al. (2024), we utilized several standard metrics: BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and BertScore (Zhang et al., 2020).

However, BLEU scores generally evaluate the similarity and identification of word spans. By nature, these scores tend to be higher in some generation tasks where the generated texts are more consistent, and lower in others. In this paper, we observe the same trend: the BLEU scores for motion descriptions 2 are higher than those for generated instructions 3. Model reliability is not the reason for this difference. The human evaluation in 5 further supports the performance superiority of the proposed model.

Therefore, to better assess the quality of generated text instructions, we incorporated G-Eval (Liu et al., 2023b) to evaluate the consistency between the predicted instructions and the ground truth. The scale was between 1 to 5. This was done by prompting Claude to evaluate the score of consistency with a dedicated template as follows (See. Sec E.9). We followed the original paper and prompted Claude five times to avoid ties in scores. Additionally, we consulted domain experts to identify subtle differ-

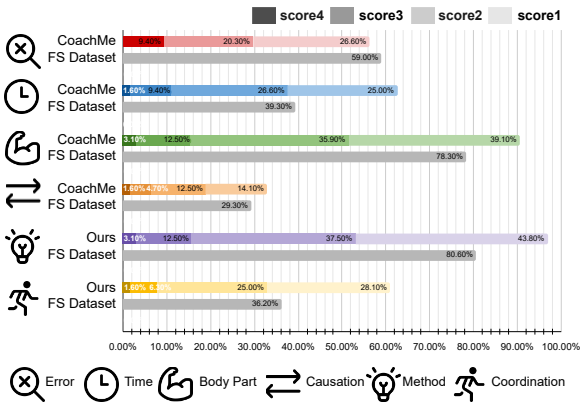

Figure 6: The horizontal bar chart represents the total number of predicted figure skating instructions of CoachMe assigned to each indicator (Sec. 4.4), with colors indicating their respective G-eval scores. The gray bar is the ground truth boxing instruction in FS dataset.

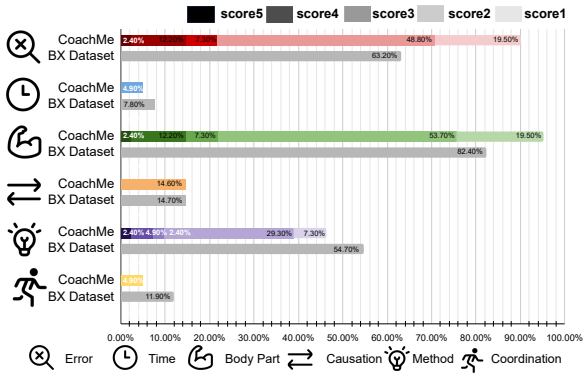

Figure 7: The horizontal bar chart represents the to- tal number of predicted boxing instructions of CoachMe assigned to each indicator (Sec. 4.4), with colors indicating their respective G-eval scores. The gray bar is the ground truth boxing instruction in BX dataset.

ences that can be discerned only by specialists in the field.

E.2 Detailed Overview of Six Sport Indicators

In Section 4.4, we define six key indicators, each of which is described in detail below. The evaluation of whether the instruction contain these factors is based on the design of prompts in reference to the G-eval template (Liu et al., 2023b), as detailed in Section E.9.

(1) Detecting errors in the motion – Identifying mistakes such as improper posture, imbalance, or misalignment.

(2) Identifying timing information – Determining if an error occurs **during takeoff, mid-air, or landing**, or if there is a **rhythm disruption**.

(3) Recognizing body part movements – Pointing out the specific body parts involved in an error, such as the left knee dropping or the right shoulder tilting.

(4) Identifying causal relationships in the sport – Explaining how an error affects performance, such as instability from poor foot positioning.

(5) Explaining how to improve the sport – Providing corrections such as lifting the right leg higher for better balance.

(6) Describing how body parts coordinate or interact – Explaining how multiple body parts should work together or interact, such as aligning the hips and shoulders during rotation.

E.3 Sport Indicators on FS and BX datasets

In addition to analyzing the instructions predicted by CoachMe, LLaMA, and GPT-4o, we also examined the distribution of sport indicators in the ground-truth instructions annotated by professional coaches. The differences in sport indicator distributions between figure skating and boxing, as shown in the Figure 6 and Figure 7, highlight the significant distinctions between these two sports. In both figures, we can also observe that the distribution of instructions predicted by CoachMe is almost consistent with those of FS and BX.

CoachMe achieves G-eval consistency scores of 2.20 and 1.83 with respect to the ground-truth instructions in the figure skating (FS) and boxing (BX) datasets, respectively. These scores indicate that, while there remains a noticeable gap in semantic similarity to the ground-truth instructions, CoachMe successfully captures and aligns with the sport-specific instructional patterns. This is evident in its sport indicator distributions, which closely mirror the domain-specific patterns in both sports. These results suggest that CoachMe is capable of generating sport-specific instruction.

We also use six sport indicators to analyze our ground truth instruction in the FS and BX datasets, as shown in Figure 8. The numbers in Figure 8 indicate the frequency with which the guidance prompts contain the indicators corresponding to both the row and the column.

$$P = \frac{\text{number of instructions containing two indicators}}{N}, \tag{11}$$

where N denotes the total number of instructions. After comparison with Figure 5, we can find that

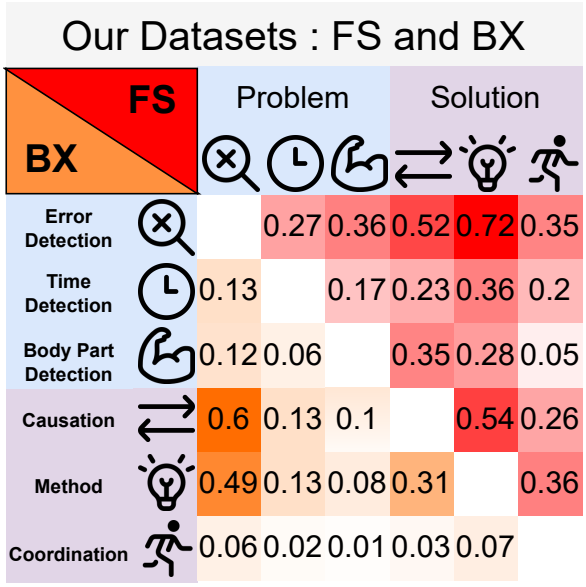

Figure 8: Effectiveness of indicator combinations on the FS and BX datasets. In the evaluation matrix, red numbers in the upper-right triangle represent scores on the FS dataset. Orange numbers in the lower-left triangle show scores on the BX dataset. Each number denotes the percentage P of the ground truth that contain two indicators, calculated based on Eq. 11

the distribution of the FS and BX datasets shows a striking similarity to the distribution of instructions predicted by CoachMe for figure skating and boxing videos.

E.4 Comparison of Modalities in Human Evaluation

In this paper, we also collaborate with figure skating coaches to provide professional evaluations comparing the three configurations of our model: Basic CoachMe, CoachMe (RGB), and CoachMe (Baseline). Table 15 illustrates the results. Basic CoachMe and CoachMe (RGB) settings fail to provide key insights for improving jumps. They frequently generate guidance that always applies, such as "Try to keep your body straight while in the air," while overlooking other more critical issues. Instead, CoachMe provides precise and personalized instructions. Notably, CoachMe generates instructions most frequently rated as the best. These results align with the quantitative evaluation results shown in Table 5. It also validates that using skeleton as the motion token modality achieves superior performance in skating.

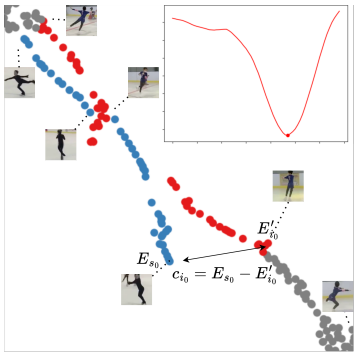

Figure 9: Target motion retrieval (red) from input video using reference video (blue). Top right: alignment cost for each starting frame. Gray dots indicate frames outside the selected segment.

Reference type	Best (%)	Worst (%)
Basic CoachMe	28.2	**25**
CoachMe	**48.4**	36
CoachMe (RGB)	23.4	39

Table 15: Win rate of each CoachMe setting in human evaluation.

E.5 Sport Indicators and Negative Factors in Human Evaluation

To further analyze the effectiveness of generated instructions, during human evaluation by coaches, when a coach rated a model-generated instruction as the best or worst, they were also asked to specify the reasons for their selection, as shown in Figure 11. The six reasons (See Sec. 4.4 and Sec. E.2) why an instruction is rated as best correspond exactly to the six sport indicators. Therefore, we can examine the key indicators that contributed to high-quality feedback in human evaluation. The four reasons why an instruction is rated as worst are the following below:

(1) General – The instruction is too general.

(2) Irrelevant – The instruction is irrelevant to the movement.

(3) Incorrect Body Part – The instruction does not identify the body part correctly.

(4) Incorrect Time – The instruction does not identify the time information correctly.

Professional coaches are allowed to select more than one reason for rating an instruction as best or worst if multiple reasons apply, or to choose none if none are appropriate.

E.6 Analysis of Human Evaluation on Figure Skating

We analyzed the proportion of each sport indicators when a model was rated good, revealing how different models emphasize key instructional elements. Table 16 presents the distribution of 6 sport

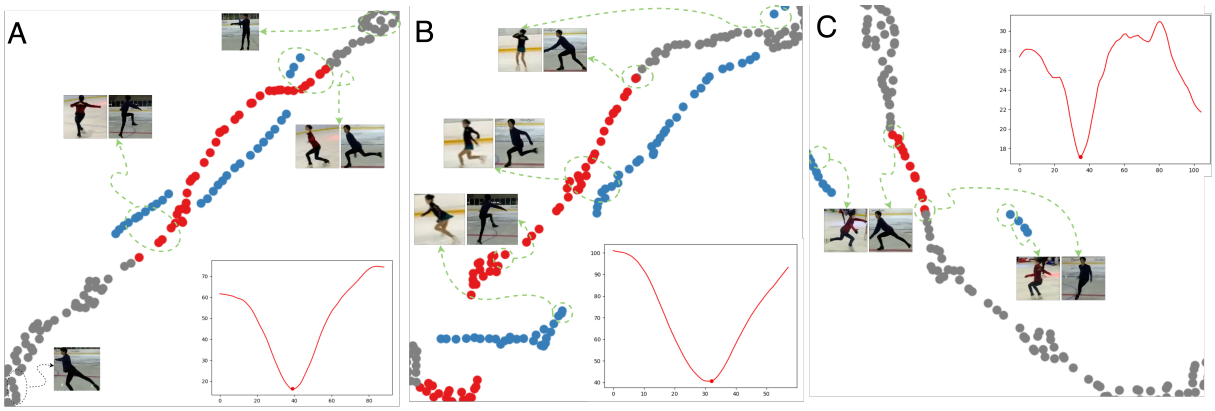

Figure 10: When possible, we mark the same frame using a single circle and link it to the frame comparison. If this is not possible, two circles are drawn to indicate their corresponding frames.

reasons	GPT-4o	LLaMA	CoachMe
Figure Skating (FS)			
Rated as Best (%)	31.3	35.9	32.8
Error (%)	12.5	10.9	10.9
Time (%)	12.5	14.1	20.3
Body Part (%)	9.4	14.1	18.8
Causation (%)	17.2	15.6	18.8
Method (%)	15.6	10.9	21.9
Coordination (%)	3.1	4.7	12.5
Rated as Worst (%)	34.4	31.3	34.4
General (%)	23.4	35.0	25.0
Irrelevant (%)	10.9	18.8	12.5
Incorrect Body Part (%)	9.4	14.1	17.2
Incorrect Time (%)	4.7	6.3	3.1
Boxing (BX)			
Rated as Best (%)	26.8	39.0	34.1
Error (%)	42.1	71.4	56.5
Time (%)	31.6	57.1	43.5
Body Part (%)	21.1	47.6	39.1
Causation (%)	36.8	38.1	26.1
Method (%)	21.1	38.1	30.4
Coordination (%)	31.6	57.1	21.7
Rated as Worst (%)	39.0	39.0	22.0
General (%)	60.0	40.0	44.4
Irrelevant (%)	33.3	20.0	22.2
Incorrect Body Part (%)	53.3	20.0	11.1
Incorrect Time (%)	0	0	0

Table 16: Comparison of the proportions of best/worst reasons for each model on Boxing (BX) dataset, based on 41 test videos and their corresponding predicted instructions. The percentages indicate how often an instruction was rated as best or worst due to the presence of the corresponding sport indicators or reasons. Error denotes error identification. Time denotes timing recognition. Body Part denotes body part awareness. Causation means causal relationships. Method means corrective methods. Coordination means coordination analysis. (See Sec. 4.4 and Sec. E.2).

indicators across GPT-4o, LLaMA, and CoachMe on Figure Skating (FS) dataset. Among the six indicators, CoachMe consistently demonstrated superior performance, particularly in timing recognition (20.3%), body part awareness (18.8%), causal relationships (18.8%), and corrective methods (21.9%). This suggests that CoachMe not only identifies movement errors but also provides more actionable and context-aware feedback. Notably, coordination analysis, which is crucial for complex sports movements, was significantly higher in CoachMe (12.5%) compared to GPT-4o (3.1%) and LLaMA (4.7%).

Regarding the quality of the generated instructions, we consulted professional figure skating coaches to obtain expert assessments. Overall, the coaches expressed strong appreciation and were notably impressed by the quality of instructions produced by CoachMe. Despite this positive reception, we conducted a further analysis of the subset of instructions (29.7%) that received "bad" ratings in the human evaluation, as shown in Table 5.

After discussions with coaches, we identified three common issues that contributed to the rating of an instruction as low quality on CoachMe.

(1) Misidentification of jump types (30.4%)**:** CoachMe identifies errors related to a specific jump type-for example, describing a toe loop when the actual jump performed is an Axel.

(2) Correct but generic instruction (21.7%)**:** The predicted instruction is technically accurate but fails to highlight the most critical technical detail that requires correction.

(3) False positive instructions (30.4%)**:** The instruction correctly identifies a specific error, but the athlete's actual execution is technically sound,

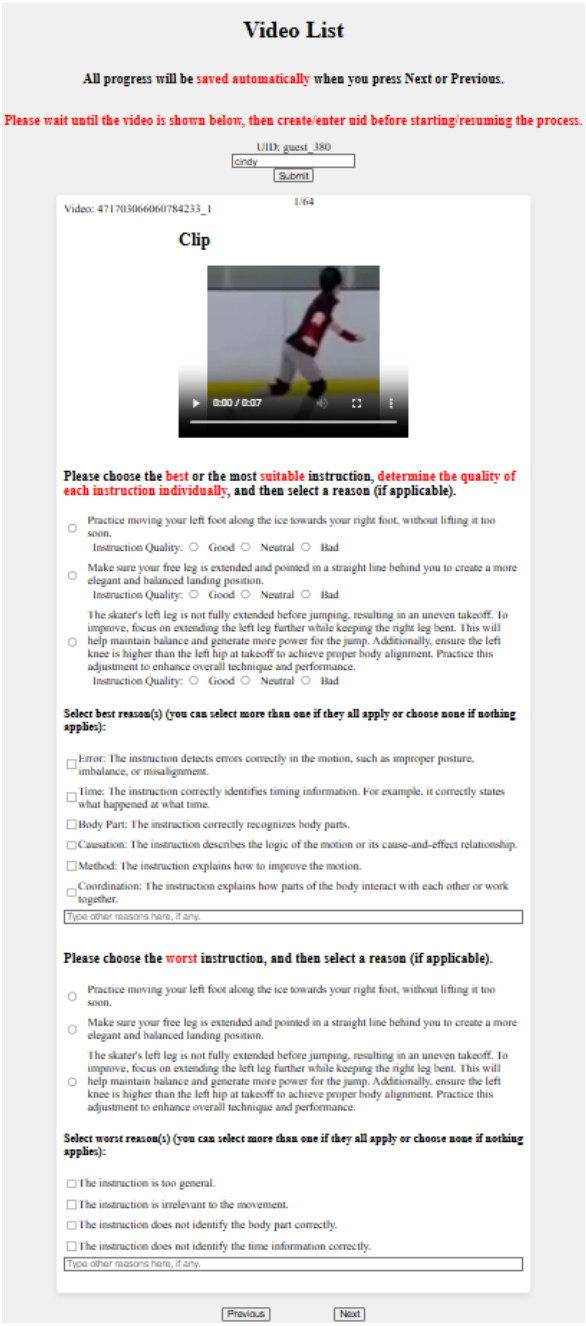

Figure 11: A sample video and its generated responses from three different models shown on the 5.3.

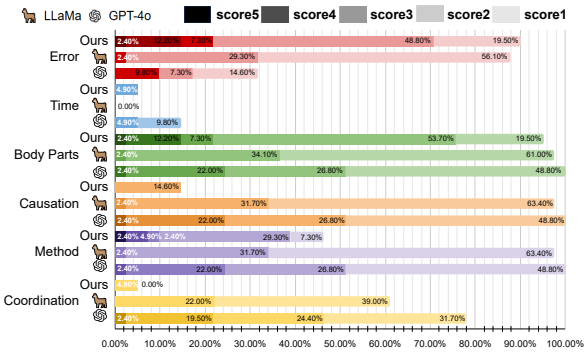

Figure 12: The horizontal bar chart represents the total number of instructions assigned to each indicator (Sec. 4.4), with colors indicating their respective G-eval scores. Higher G-eval scores indicate more informative and effective guidance that aligns more closely with the ground truth. The G-eval scores range from 1 to 5.

leading to unnecessary corrections.

The first issue arises from our pretrained model trained on the HumanML3D dataset, which lacks detailed examples of specialized figure skating jumps. While these jumps are technically distinct, their biomechanical similarities make them difficult to differentiate. This limitation is more pronounced in general models like GPT-4o and LLaMA, which are not designed to capture domain-specific biomechanical cues.

The occurrence of the second issue highlights the need for incorporating sport-specific knowledge into the instruction generation process. This problem is more prevalent in GPT-4o and LLaMA that often provide broadly accurate yet overly generic instructions, missing critical nuances (see Section 6.1 for detailed analysis). While the design of CoachMe aims to alleviate these issues, certain domain-specific inconsistencies persist. This highlights the need for further refinement of our specialized models to more accurately identify and convey the most relevant movement details.

The third issue stems from a bias in our dataset, where all ground-truth instructions focus on suggesting improvements to the performed actions. This distribution leads CoachMe to implicitly assume that every input requires correction, even when the performance is already acceptable. To remedy this, we plan to introduce a scoring mechanism and integrate examples of high-quality movements with positive annotations. While related to the general challenge of hallucinations in large language models, we believe this issue can be effectively mitigated through future enhancements to

our training data and model design.

E.7 Analysis of results on Boxing

The boxing coach who conducted the human evaluation also noted that since each clip features only a basic punch and is very short, temporal errors are not relevant. This explains why the "Incorrect Time" factor in Table 16 consistently accounts for 0% of the negative ratings. As shown in Figure 12, We also observe that, across all models, instructions perform unusually poorly on the sport indicator "Time", suggesting that when the video is short—such as containing only a single punch—none of the models are able to provide meaningful temporal descriptions. This issue persists even in CoachMe, which specifically includes a Concept Difference Module designed to handle temporal information.

Despite this limitation, CoachMe outperforms both LLaMA and GPT significantly in the boxing domain across NLP metrics, G-eval, and human evaluations.

However, as shown in Figure 12, CoachMe performs relatively poorly on the sport indicator Coordination. To understand this, we consulted boxing experts, who pointed out that in current boxing videos, it is often sufficient to describe individual body parts rather than the relationships between them. As a result, instructions generated by CoachMe can still receive high scores even with a low mention rate of Coordination.

This is because for beginner-level learners in the BX dataset, it is more effective to provide instructions targeting single body parts rather than describing complex interrelations among different body parts. Overloading beginners with corrections involving multiple body parts and their coordination may increase the complexity of the movements, making them harder to execute correctly. Therefore, professional coaches tend to focus on adjusting one specific body part at a time. This is particularly relevant in fundamental actions such as "Jab" and "Cross" or in stance training, which are designed to help novice boxers understand how to protect themselves during actual sparring.

This observation is reflected in the low frequency of the sport indicator "Coordination" in both the ground-truth distribution of the BX dataset and the instruction predictions generated by CoachMe, as shown in 7 and 8. Despite incorporating a Human Pose Perception designed to model coordination of different body parts, CoachMe effectively learns this pattern—providing simple, targeted instruction that mirrors such coach style employed by professional coaches when training beginners.

This highlights the importance of domain-specific knowledge, as each sport involves unique movement patterns. This is precisely the design philosophy behind CoachMe: the ability to adapt to different sports by effectively acquiring domain-specific knowledge, supported by lightweight task-specific adaptation models. However, this also suggests that the instructions should not only be sport-specific, but the evaluation indicators, the sport indicators, may also need to be tailored for each sport. In the future, it may be beneficial to design sport-specific sets of indicators tailored to the characteristics of each sport and action type.

E.8 Human Evaluation Interface

Figure 11 demonstrates the human evaluation interface used in the experiment. Our annotator is required to select the best and worst answers generated by the models, providing an explanation for each choice. Note that throughout the evaluation, the annotator is not provided any additional information (e.g., ground truth). This ensures that the information seen by the annotator aligns with the information available to our model.

E.9 G-Eval Template

We follow the original G-Eval paper (Liu et al., 2023b) to design tailored prompts for evaluating the generated instructions. Claude (via its API) is employed as the evaluation LLM in our implementation of the G-Eval methodology. We adopt the G-Eval "Consistency" template to assess how well the generated instructions align with the ground truth, as illustrated in Table 17. We also design specific prompts to evaluate the quality of instructions based on the indicators we have defined, as described in Section 4.4. We adopt 6 G-Eval templates to evaluate specific aspects of instructional content: error detection (See Table 18), time-related expressions (See Table 19), descriptions that mention specific body parts (See Table 20), causation or logical explanations of motion (See Table 21), methods for improvement (See Table 22), and coordination between body parts (See Table 23).

You will be given an instruction provided by the coach. You will then be given one rephrased version of that instruction.

Your task is to rate the rephrased version on one metric.

Evaluation Criteria:

Consistency (1-5) - the factual alignment between the summary and the summarized source. A factually consistent rephrased version contains only statements that logically follow from the original instruction.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Read the rephrased version and compare it to the coach's original instruction. Check if the rephrased version contains any factual inaccuracies that are not supported by the original instruction.
3. Assign a score for consistency based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Document}}

Rephrased Version:

{{Summary}}

Evaluation Form (scores ONLY):
Consistency: [Insert score here]

You only need to give a score of this example directly.

Table 17: Consistency Template

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Error Detection (0-1) - The instruction contains any wording that can point out the error clearly.
0: The instruction doesn't clearly point out the error.
1: The instruction clearly points out the error.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that can point out the error clearly. Ensure that the athlete understands exactly what the problem is after hearing the instruction. Just like these two coaching instructions: "Make sure not to position your right leg too far behind your left leg when preparing for takeoff." and "Your first jump is good. For the second jump, avoid overturning your left side before takeoff and bend your right knee more for better height."
3. Assign a score for Error Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}
Evaluation Form (scores ONLY):
Error Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 18: G-eval : Error Detection Template.

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Time Detection (0-1) - The instruction contains wording that clearly includes time-related information.for example like: when take off, during the landing...
0: The instruction doesn't mention time-related information.
1: The instruction clearly mentions time-related information.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that clearly includes time-related information. Ensure that the athlete can understand when and how to act after hearing the instruction. Just like these three coaching instructions: "Remember to keep your posture straight and vertical during the initial stages of the jump, it will help in achieving the correct form.", "Bend your hands a bit at takeoff to aid your spin." and "Stretch your right side backward and left side forward beforehand to increase power."
3. Assign a score for Time Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}

Evaluation Form (scores ONLY):
Time Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 19: G-eval : Time Detection Template

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Body Parts Detection (0-1) - The instruction contains any wording that includes body parts clearly.
0: The instruction doesn't clearly point out the body parts.
1: The instruction clearly points out the body parts.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that can point out the body parts clearly. Ensure that the athlete understands exactly which body parts to focus on after hearing the instruction. Just like these two coaching instructions: "Practice moving your left foot along the ice towards your right foot, without lifting it too soon." and "Keep your right leg from going over your left leg during takeoff for a smoother jump. Ensure your right toe lock stays firm. Manage your hands, don't lift them too high while spinning, and make a circle with them."
3. Assign a score for Error Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}

Evaluation Form (scores ONLY):
Body Parts Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 20: G-eval : BodyPart Detection Template

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Causation Detection (0-1) - The instruction contains any wording that can describe the logic of the motion or its cause-and-effect relationship clearly.
0: The instruction doesn't clearly point out the cause-and-effect relationship of the motion.
1: The instruction clearly points out the cause-and-effect relationship of the motion.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that can describe the logic of the motion or its cause-and-effect relationship clearly. Ensure that the athlete understands exactly what the logic of the motion is after hearing the instruction. Just like these two coaching instructions: "Practice keeping your left leg closer to your right foot and lifting your left knee higher to help create more power for a higher jump because you are not fully rotating your jumps right now." and "Remember to keep your posture straight and vertical during the initial stages of the jump, it will help in achieving the correct form."
3. Assign a score for Error Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}

Evaluation Form (scores ONLY):
Body Parts Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 21: G-eval : Causation Detection Template

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Method Detection (0-1) - The instruction contains any wording that can clearly explains how to improve the motion.

0: The instruction doesn't clearly explains how to improve the motion.
1: The instruction clearly explains how to improve the motion.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that can clearly explains how to improve the motion. Ensure that the athlete understands exactly what the logic of the motion is after hearing the instruction. Just like these two coaching instructions: "Practice moving your left foot along the ice towards your right foot, without lifting it too soon." and "You need to have more speed before the jump."
3. Assign a score for Error Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}

Evaluation Form (scores ONLY):
Body Parts Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 22: G-eval : Method Detection Template

You will be given an instruction provided by the coach.

Your task is to rate the instruction on one metric.

Evaluation Criteria:

Coordination Detection (0-1) - The instruction contains any wording that can clearly explains how the body coordinates movements or how more than two body parts work together.
0: The instruction doesn't clearly explains how the body coordinates movements or how more than two body parts work together.
1: The instruction clearly explains how the body coordinates movements or how more than two body parts work together.

Evaluation Steps:

1. Read the coach's instruction carefully and identify the main facts and details it presents.
2. Check if the instruction contains any wording that can clearly explains how the body coordinates movements or how more than two body parts work together. Ensure that the athlete understands exactly how to coordinate the movement after hearing the instruction. Just like these two coaching instructions: "Practice moving your left foot along the ice towards your right foot, without lifting it too soon." and "Make sure not to position your right leg too far behind your left leg when preparing for takeoff. Stretch your right side backward and left side forward beforehand to increase power. Try using more force for the takeoff and rotate your body towards the jump to raise your left leg higher."
3. Assign a score for Error Detection based on the Evaluation Criteria.

Example:
Coach's Instruction:

{{Instruction}}
Evaluation Form (scores ONLY):
Body Parts Detection: [Insert score here]

You only need to give a score of this example directly.
ONLY OUTPUT A SINGLE NUMBER (0 or 1) WITH NO ADDITIONAL TEXT.

Table 23: G-eval : Coordination Detection Template