AgentCoach: LLM-Based Adaptive Coaching Feedback for Motor Skill Learning

Dizhi Ma*
Elmore Family School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, USA
ma742@purdue.edu

Jiakun Yu*
Department of Computer Science
Purdue University
West Lafayette, Indiana, USA
yu1591@purdue.edu

Xinyi Wang
Edwardson School of Industrial Engineering
Purdue University
West Lafayette, Indiana, USA
wang6185@purdue.edu

Xiyun Hu
School of Mechanical Engineering
Purdue University
West Lafayette, Indiana, USA
hu690@purdue.edu

Liang He
Department of Computer Science
The University of Texas at Dallas
Richardson, Texas, USA
liang.he@utdallas.edu

Sooyeon Jeong
Department of Computer Science
Purdue University
West Lafayette, Indiana, USA
sooyeonj@purdue.edu

Karthik Ramani
School of Mechanical Engineering
Purdue University
West Lafayette, Indiana, USA
Elmore Family School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, USA
ramani@purdue.edu

Abstract

We present AgentCoach, an LLM-powered system that provides adaptive feedback for motor skill learning from tutorial videos. The system works by extracting key coaching points (CPs) and compiling CP-specific evaluators that map each cue to measurable kinematic parameters. This process allows AgentCoach to connect high-level semantic meaning with low-level postural estimation for accurate, context-aware evaluation. During practice, learners receive concise visual diagnostics of their mistakes paired with prescriptive verbal feedback that adapts based on their performance history. We technically validate the CP extraction and evaluator compilation across a wide range of common sports and exercise videos. A user study confirms the system's usability and shows the system's potential effectiveness of its adaptive feedback across multiple skills.

CCS Concepts

• **Human-centered computing** → **Natural language interfaces**; • **Computing methodologies** → *Natural language generation*; • **Applied computing** → Interactive learning environments.

Keywords

Motor Learning, Adaptive Coaching, Interactive Learning Systems, Large Language Model

ACM Reference Format:
Dizhi Ma, Jiakun Yu, Xinyi Wang, Xiyun Hu, Liang He, Sooyeon Jeong, and Karthik Ramani. 2026. AgentCoach: LLM-Based Adaptive Coaching Feedback for Motor Skill Learning. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 21 pages. https://doi.org/10.1145/3772318.3791652

*Equal contribution.

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/26/04
https://doi.org/10.1145/3772318.3791652

1 Introduction

Motor skill acquisition is fundamental, whether in athletic training, such as tennis or weight lifting, or in daily movement practices, such as Tai Chi and Yoga. Since high-quality instructions and feedback shape learning outcomes for learning new motor skills [59, 88], many sports trainees rely heavily on human expert coaches' and instructors' in-person feedback through repetitive practice and training sessions. Typically, these coaches observe the trainee's movement/execution, compare it against standards, and provide adaptive feedback based on the learner's progress, misalignment, and individual needs. However, access to human coaching is often constrained by cost, geography, and availability.

In order to address this gap, some researchers have developed postural visualization systems to support motor skill learning in sports [45, 49, 79], rehabilitation [22, 30], and general exercise [3, 53]. These systems often compare the user's pose (sequence) with a reference one from experts to locate the error joints or limbs.

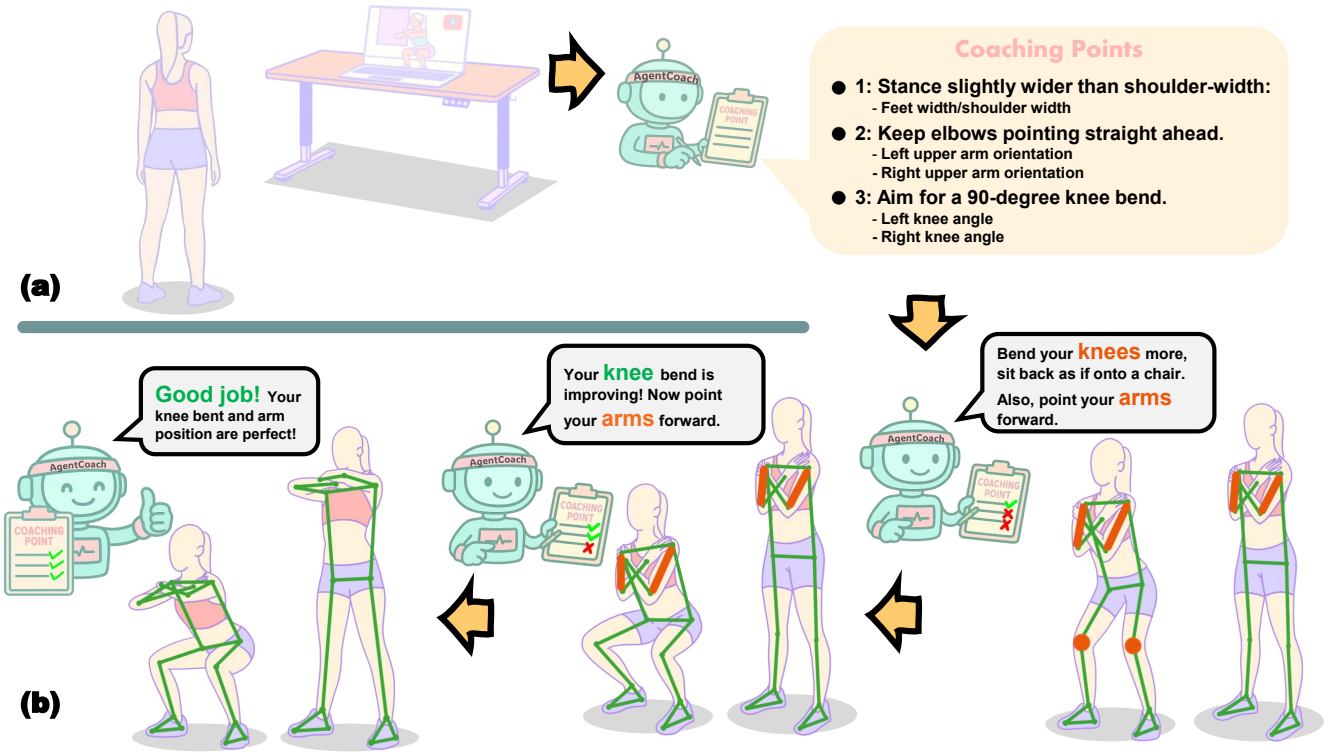

Figure 1: AgentCoach: an LLM-powered motor-learning system that turns tutorial videos into actionable coaching. (a) From a tutorial video, it extracts key coaching points (CPs) and assesses the learner's motion on each CP, linking low-level pose deviations to high-level semantics. (b) Using CP-wise results over time, it tracks progress and delivers progress-adaptive verbal feedback with visual overlays.

Based on this comparison, they visualize the differences between the correct postures and the extracted user postures by showing the wrong body parts and the correct postures. Such feedback could serve as part of the coach's duties, which is to point out the mistakes and demonstrate the correct ones. Nevertheless, coaches not only **show**, but also **tell**. Showing refers to demonstrating the correct movement and visually indicating deviations, helping learners see what their posture should look like [51, 64]. Telling, in contrast, involves articulating the underlying coaching points (i.e., the critical features or key elements of a skill emphasized by coaches [26, 60]). Coaching points (CPs) highlight which biomechanical aspects matter, why they matter, and how the learner should adjust (e.g., "bend your knees more" or "rotate your hip to generate power") [52, 85]. This distinction highlights that, beyond demonstration, coaching also relies on explicit, semantically meaningful guidance that connects observed errors to corrective principles. Recent work has also attempted to structure feedback around point-wise evaluations. For example, Weng et al. [83] proposed a knowledge-based standard operating procedure (SOP) framework that breaks down basketball shooting into step-level postural points and delivers binary SOP results (i.e., pass or not). However, the feedback is delivered by a coach-in-the-loop (Wizard-of-Oz) protocol rather than an explicit algorithmic mapping from pose to CP-wise prescriptions. Automating this mapping is necessary for consistent and scalable guidance, and for logging CP-wise results that enable prescriptive feedback.

Yet even with CP-wise feedback that articulates what to correct, effective coaching is also about when and how to adjust guidance across a learner's journey. Expert human coaches naturally adapt their feedback based on each learner's performance history, offering corrections [5, 27] for initial attempts, giving motivational praise [87] when improvements are made, and delivering brief reminders [40, 87] or simply staying silent to reduce guidance effect [84]. Regardless of the learner's prior attempts or error patterns, existing motor learning systems deliver non-adaptive feedback—for example, identical visualizations [44, 86], the same vibration pulses [7, 75], or fixed auditory hints [89, 90]—whenever a deviation is detected. These one-size-fits-all approaches are not able to reproduce personalized, adaptive instruction as human coaches. While studies have shown the benefits of multimodal feedback, there has been limited investigation into progressive feedback adaptation that mimics the contextual behaviors of human coaches.

Recent advances in large language models (LLMs) present an unprecedented opportunity to address these limitations. Both commercial tools [28, 56] and recent research [95] demonstrate that multimodal LLMs can reliably extract hierarchical structures and key teaching points from instructional videos, converting them into

structured notes that capture both visual and verbal cues. This indicates that extracting CPs from sports or exercise tutorials—where domain experts similarly emphasize critical poses and CPs—is feasible with current multimodal capabilities. Meanwhile, emerging works [4, 50] have begun to explore directly generating expert feedback from video. However, their mechanisms—taking videos as input and producing one-shot text feedback—remain opaque in how feedback is explicitly grounded in gold-standard posture targets (e.g., CPs) and coaching principles. Taken together, these developments reveal the potential of LLM/VLM-powered systems to generate contextual, CP-level feedback that more closely mirrors human coaching.

Therefore, we propose AgentCoach, a multimodal LLM system that combines traditional visual pose feedback with adaptive verbal coaching based on CP-wise evaluation. Motivated by our formative study with experienced coaches and learners from common sports and exercises, the system aims to provide progress-adapted CP-centric instruction. To support the CP-wise analysis, we proposed a CP-to-parameter mapping taxonomy via coding of 55 online tutorial videos. With this taxonomy, AgentCoach extracts key CPs and reference movements from instructional videos, maps CPs' semantic description into measurable parameters, and delivers multimodal feedback. While the visual component follows established approaches by highlighting postural errors, the verbal component bridges the visual highlights to actionable CPs and introduces adaptive strategies that adjust feedback content and timing based on individual practice history. We validated AgentCoach's CP extraction and CP-wise evaluator building functionalities through a numerical study with tutorial videos. A two-phase study was conducted with 24 participants: (1) comparing visual-only feedback, visual + fixed corrective verbal feedback, and visual + adaptive CP-wise verbal feedback to isolate the contribution of our adaptive coaching approach, and (2) evaluating usability and exploring its potential to support motor skill improvement across six different motor skills (e.g., tennis, squat, break dance, etc). In summary, we contribute:

- A mapping library that bridges high-level instructions to quantifiable low-level pose-based measures, supporting CP-centric design goals derived from formative study.
- AgentCoach, an LLM-powered system that combines traditional visual pose feedback with adaptive verbal coaching based on CP-wise evaluation and performance history.
- A quantitative study evaluating the system's CP extraction and parameter mapping capability, and a two-phase user study demonstrating the usability and potential efficacy of our adaptive feedback approach across multiple motor skills.

2 Related Work

2.1 Posture- and Computer Vision-based Motor Learning Systems

With the development of computer vision, researchers have adopted pose estimation algorithms [9, 38, 48, 80] to analyze images and videos. Such methods predict the positions of key joints in each frame, for example, wrists, shoulders, hips, knees, and facial landmarks. By linking these joints, angles and orientations can be directly computed.

In the motor learning domain, researchers have been leveraging these pose estimation tools to analyze and visualize learners' performance in order to locate misalignments. Previous work in sports training systems [36, 45, 49, 72, 79, 82], rehabilitation tools [22, 30], and general exercise applications [3, 13, 74, 76, 97] often relied on such pose estimation approaches to provide demonstrative or comparative visualization as the primary mechanism for feedback. For example, Liu *et al.* [45] developed PoseCoach, which allows users to compare running posture and provides suggested viewpoints for better understanding. Escalona *et al.* [22] proposed EVA, an augmented reality platform for at-home rehabilitation sessions. Anderson *et al.* [3] introduced YouMove, which used Microsoft Kinect to estimate a user's pose and embed it within an augmented mirror-based system. Wen *et al.* [82] also used data captured by Microsoft Kinect to reconstruct trainees' motions in 3D, allowing the coaches to review and comment from arbitrary viewpoints. Commercially, the Kemtai platform[1] also adopts vision-based pose estimation, offering real-time corrective feedback for at-home physical therapy and rehabilitation. However, regardless of the medium or scenario, most of these systems rely on pure posture-based comparison with pre-recorded reference motions and provide limited semantic interpretation of how or why to adjust movements. Users can only compare their pose against the reference, but lack guidance about the meaning of the misalignment or how to correct it.

Outside of research systems, most learners turn to online tutorial resources on platforms such as YouTube, Instagram, and TikTok. While highly accessible, such learning experiences remain passive due to the absence of feedback. Researchers have begun to transform online video tutorials into more interactive and immersive experiences [13, 35, 42]. For instance, Video2MR [35] extracts 3D human motions from YouTube videos and renders them as avatars for mixed reality (MR) visualization. Although this lowers the barrier to creating immersive learning content, such approaches still focus solely on pose information without leveraging the rich semantic layer that naturally accompanies the videos.

Beyond posture comparison, recent work in basketball has introduced a knowledge-based Standard Operating Procedure framework [83], which encodes expert coaching insights into step-by-step posture checks. This approach demonstrates the value of combining pose estimation with structured coaching knowledge rather than relying solely on pose similarity. However, its feedback currently depends on human evaluation (via a Wizard-of-Oz method [25, 39]) and provides binary (correct/wrong) feedback rather than how to adjust, and thus does not yet fully bridge the gap between low-level pose information and high-level semantic meaning.

In this work, we aim to bridge pose estimation with semantic interpretation, democratizing the learning experience by transforming online tutorials into motion references and actionable guidance.

2.2 Human Coach Behavior

Sport science and HCI describe coaching feedback in terms of what content is delivered, how timing and frequency are scheduled, and the learner's stage in skill acquisition [55, 68, 91]. In practice, coaches usually decompose the skill into elements and guide

[1] https://kemtai.com/

trainees to practice. Prior work [47, 73] distinguishes an initial acquisition period, a middle period with unstable control, and a later period focused on refinement.

At the beginning, learners are establishing the movement pattern. Coaches tend to use concrete cues anchored to observable checkpoints [5, 27] (e.g., establish a stable posture; complete a specific transition). When a misalignment keeps recurring, coaches provide plain-language clarification or a brief demonstration so that the next attempt is guided by a clear change [5, 32]. Feedback is relatively frequent to reduce drift from the intended pattern, and correct attempts are acknowledged briefly to stabilize the emerging form [55, 68].

In the middle period, the basic practice pattern is present but inconsistent: lapses occur intermittently rather than on every attempt. Coaching shifts from shaping to stabilizing [27, 47]. When attempts are consistently correct, coaches avoid trial-by-trial comments; occasional brief confirmations are sufficient [32, 67]. When a deviation appears, a timely reminder is provided so the correction is linked to the error instance, which supports longer-term retention without creating dependence on continuous cues [1, 61].

In the refinement period, feedback becomes infrequent [55, 68]. Coaches comment only when a clear deviation appears, otherwise using reduced-frequency or summary feedback to avoid unnecessary reliance on coach-delivered cues [55]. With greater stability, interaction emphasises learner initiative and motivation, using concise corrections only when warranted [21].

Beyond coach-initiated input, prior work highlights benefits of *self-coaching*: selectively requesting information when needed, keeping simple notes on recent attempts, and running quick self-checks. Studies on learner-controlled timing and longer-term observations suggest this active role supports engagement and can yield small, task-specific gains [21, 63, 69]. Guided by these findings and our formative study, we design *AgentCoach* to be CP-based, providing adaptive, progressive feedback.

2.3 LLMs and VLMs for Motion Understanding

Recent advances in large language models (LLMs) and vision language models (VLMs) have opened new possibilities across diverse research areas, such as MR interaction [15, 34, 98], education [46], design [18–20], and programming [33]. Beyond these domains, researchers have begun to explore how language models can interact with human motion data, bridging textual descriptions and physical movement. For instance, PoseFix [16] and ChatPose [23] enable language-guided 3D pose correction and generation, showing the potential of text-driven human motion editing. More closely related are works that use LLMs/VLMs to understand motion directly from video or motion data. MotionLLM [11] integrates video and motion data to understand human actions, showing potential as an intelligent "fitness coach." ExpertAF [4] takes learner videos as input and generates both expert commentary and demonstration (in pose data format). T3Set [50] introduces a multimodal dataset for table tennis stroke suggestions. While these works demonstrate exciting potential, most take video as input and directly output generic commentary, leaving them ungrounded in explicit biomechanical parameters. Additionally, they primarily focus on single-shot text generation and overlook learners' performance history, resulting in one-size-fits-all suggestions. By combining semantic grounded pose analysis of movement with coach-like instructional prompts, we aim to leverage LLMs to generate progress-adaptive feedback that mirrors human strategies.

3 Formative Study

Our study began with semi-structured interviews with 8 experienced coaches and trainees (E1-E8) across different sports and exercises. The interviews were carried out either in person or online, based on the interviewee's preference. Each interview lasted around 30 minutes, and was audio-recorded and transcribed for analysis. Demographics and questions appear in Appendix A.

3.1 Interview Topics

In formative interviews, we explored the following key topics.

- General coaching/learning experience. Interviewees were asked to describe their general coaching and learning backgrounds, with emphasis on formal coaching experience.
- Feedback provided/received. We asked participants to describe the feedback they provided or received, with a focus on its content.
- Feedback intent. Participants were asked about the different communication intents behind their feedback.
- Strategy and adaptation. Participants described effective coaching strategies delivered and received, and how they adapted feedback to a learner's progress.

3.2 Insights and Findings

I1 — Break skills into coachable units. Both coaches and learners mentioned that experts consistently decompose a skill into steps or checkpoints that anchor attention. This breakdown allows coaches to analyze atomic elements and provide actionable corrections or affirmations. *"I break down the stroke into steps, for example, racket grip, preparation stance, backswing, and fore swing..."* (E5).

I2 — Multimodal delivery. Coaches combine **physical demonstration** (presenting standard, mirroring errors, gesture-based highlighting of joints/segments) with **verbal cues**. Demonstration conveys target spatial relations, ranges, and directions of movement; short imperatives (e.g., "close the wrist," "hips through") focus attention on the relevant CPs without overloading the learner. *"I mimic what they did wrong, tell them the key point to fix, and show them what they should do instead."* (E5); *"I demonstrate step-by-step with an explanation for a novice, then full strokes."* (E4).

I3 — Feedback intents. We identified six different intents of coaches' feedback: 1. **Correction:** A corrective, prescriptive cue. *"When my student does something wrong, I tell them right away."* (E4); 2. **Explanation:** A reason, mechanism, or metaphor for a repeated error. *"If a mistake keeps happening, I use a metaphor and more detail to guide them."* (E1); 3. **Reminder:** A brief cue for a known point. *"If the old issue returns—e.g., rounding my back in a deadlift—my coach just says 'back' and I know."* (E6); 4. **Praise:** Reinforcement for the learner's improvement. *"My coaching style is encouraging; I tend to compliment progress."* (E5); 5. **Confirmation:** Acknowledgment of correct form or affirmation of a corrected mistake. *"My coach confirms when I've corrected a misalignment."* (E2); 6. **Silence:**

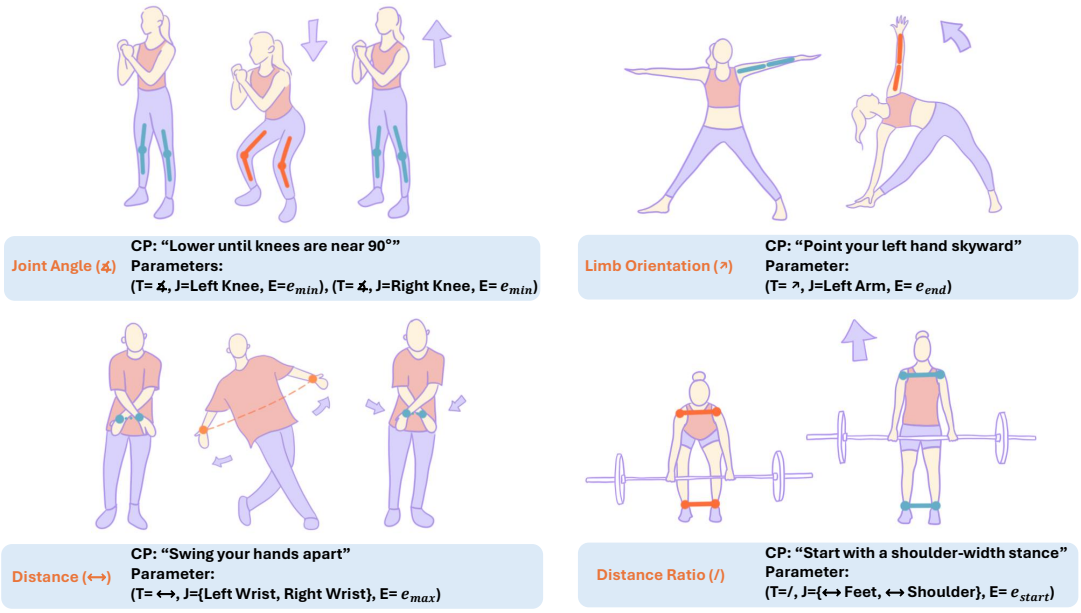

Figure 2: CP to parameter mapping examples. Note that each skill has multiple CPs; we only visualize one CP mapping for each skill in this figure.

The intentional withholding of feedback to reduce overload and encourage self-correction. This set covers most coaching turns and underpins our progression policy, with intents aligned to learner progress: early errors prompt corrections (and brief explanations), improvements invite praise/confirmations, recurring minor lapses trigger reminders, and stable performance warrants silence to avoid over-guidance.

I4 — Verbal Feedback Strategy. Coaches generally prioritize foundational (early-step) CPs before refinements, typically addressing one primary cue per attempt to avoid overload. Additionally, verbal feedback is kept concise and to the point, escalating to brief explanations or demonstrations only when an error persists. Strategy adapts to learner progress: as execution stabilizes, coaches taper from correction to reminders and confirmations, and only then introduce more advanced CPs. *"Fix the base before the details—one thing at a time."* (E3); *"Keep cues short and direct, add detail only if they repeat the mistake."* (E1).

These findings ground the design goals and the CP-to-parameter taxonomy that structure our system architecture presented next (Section 4).

4 Design Goals and CP-to-Parameter Taxonomy

4.1 Design Goals Informed by Feedback Practice

From the interview insights, we identify the following goals for AgentCoach:

DG1 — CP-centric instruction. Represent each skill as a set of CPs and align evaluation to those CPs.

DG2 — Multimodal feedback. Pair diagnostics and demonstrative visual cues with prescriptive verbal feedback.

DG3 — Progress Adapted Feedback. Adapt verbal feedback to the learner's CP-wise progress with different intents.

4.2 CP-to-Parameter Library Construction

To support CP-centric evaluation (DG1), we constructed a parameter library to link the semantic CP descriptions to measurable parameters via video coding. In this paper, we define a *coaching point (CP)* as the minimal, self-contained instruction about posture or movement that can be explicitly evaluated from body kinematics.

CP Corpus and Unit of Analysis. We collected a diverse set of **55** YouTube tutorial videos covering common sports and exercises (Appendix B). Candidate coaching points were first extracted in textual form using Gemini's [29] video understanding capability, which integrates transcribed narrations, on-screen text overlays, and visual annotations. This automatic extraction provided an initial pool of CPs aligned with the narrated and demonstrated content. To ensure coverage and quality, two researchers rewatched each video to cross-check the candidates and manually add any missing CPs, then independently reviewed the full set for accuracy and completeness. We computed Cohen's κ [14] on the CP annotations to assess reliability ($\kappa = 0.76$). Disagreements were resolved through discussion. We filtered out CPs that cannot be evaluated purely from postural information, such as those referring to muscle activation, breathing rhythm, weight distribution, and related external objects. After filtering, we obtained a total of 212 CPs, which served as the unit of analysis for subsequent taxonomy construction.

Deriving the Parameter Taxonomy . We conducted three rounds of iterative coding on the extracted CPs to derive computational

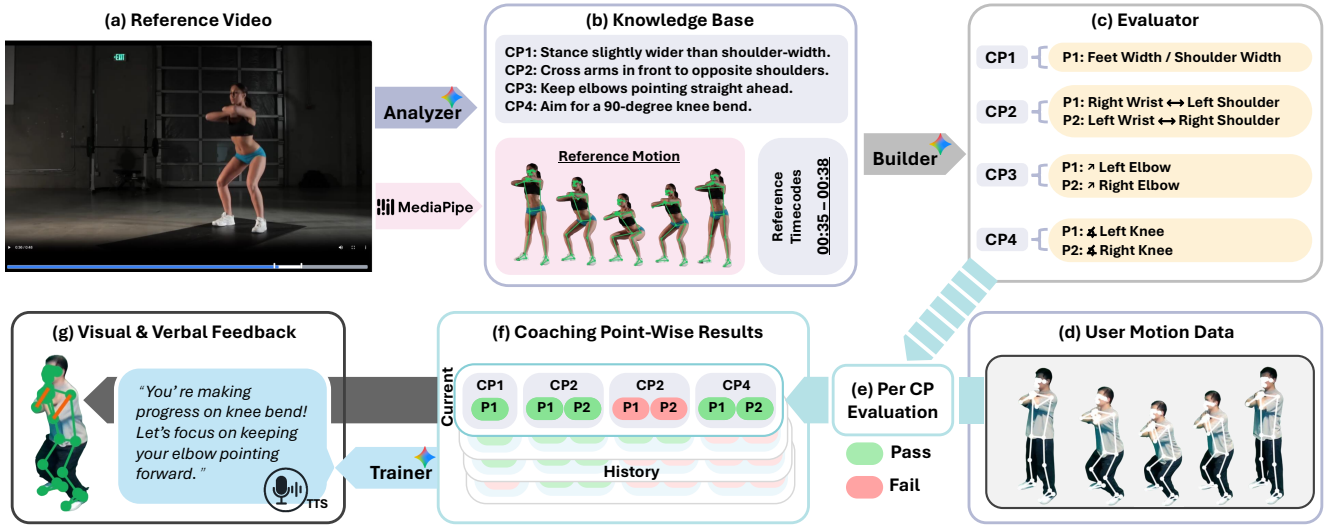

Figure 3: AgentCoach workflow. (a) Reference video input. (b) Analyzer extracts coaching points (CPs) with reference segment timecodes to extract the reference motion estimated by MediaPipe; together they form the knowledge base (KB). (c) Per-CP evaluators derived from the KB with the Builder. (d) User motion stream with performance history. (e) Per-CP evaluation on the current window. (f) Coaching Point-Wise Results (current and history). (g) Trainer agent delivers visual overlays and verbal feedback.

parameter categories. In the first round, two researchers independently performed open coding to generate a broad set of candidate parameter forms, and disagreements were resolved through discussion. In the second round, the researchers jointly refined the candidate set by merging overlapping categories, resolving ambiguous cases, and articulating inclusion rules for borderline CPs. In the final round, the refined taxonomy was applied to the full CP candidates to ensure coverage and mutual exclusivity. We also computed Cohen's κ on the independent CP-to-parameter mappings to assess the reliability of the mapping process ($\kappa = 0.73$). By the end of this process, all disagreements had been resolved, and each CP was mapped to one or more parameters under a consistent specification, yielding a complete CP-to-parameter mapping for the subsequent evaluator compilation stage.

Resulting Taxonomy. Based on our coding results, we present a *CP parameter taxonomy* that maps each CP to one or more parameter instances in a representational evaluation space $T \times J \times E$ (parameter type, joint/segment specification, temporal evaluation point). Examples are shown in Figure 2. Derived through three rounds of coding, the taxonomy converges on four main posture-centric categories: *Joint Angles* [49], defined by a vertex joint and its two adjacent joints; *Limb Orientation* [89], defined by a limb segment and its orientation; *Distances* [2], defined by a pair of joints; and *Distance Ratios* [17], defined by two joint pairs, such as stance width relative to shoulder or hip width. Each parameter is anchored to one of four *temporal evaluation points*—e_{init} (preparation), e_{max} *and* e_{min} (peak), e_{end} (ending). All parameters are instantiated using the full set of joints from MediaPipe [48] and implemented in Python.

Building on the formative insights, the design goals (DG1–DG3) and the CP parameter taxonomy together ground our system's architecture, as detailed next in Section 5.

5 AgentCoach System

The AgentCoach's Analyzer–Builder–Trainer framework consists of the following stages (Figure 3): (1) Knowledge base construction, (2) evaluator compilation, (3) user evaluation, and (4) contextual feedback generation. From a tutorial video, the **Analyzer** extracts a CP list and reference timecodes; MediaPipe estimates poses for the entire video, and the timecodes localize the reference-motion pose segment. Next, the **Builder**, augmented by our parameter library (Section 4.2), uses the CP list and reference motion to compile CP-wise evaluators (DG1). The system then performs CP-wise evaluation on the user motion, and per-CP results accumulate into a history (DG3). Finally, the **Trainer** generates history-adapted verbal feedback together with visual feedback (DG2).

5.1 Walkthrough

We now use an example of using our system. The user starts in *Learning Mode* to upload a barbell deadlift tutorial video and AgentCoach process the video. They (i) watch the tutorial clip with an optional pose skeleton overlay (Figure 4 a-1) on the coach and loop a selected segment for focused repetition (Figure 4 a-2); (ii) inspect and edit the automatically extracted coaching points (e.g., stance width), where they could optionally check out the detailed explanation and irrelevant CPs can be removed to tailor the plan (Figure 4 b); and (iii) ask free-form questions in the LLM chat (e.g., common mistakes or self-checks), which returns cue-aligned answers grounded in the selected CPs (Figure 4 c). Before practice,

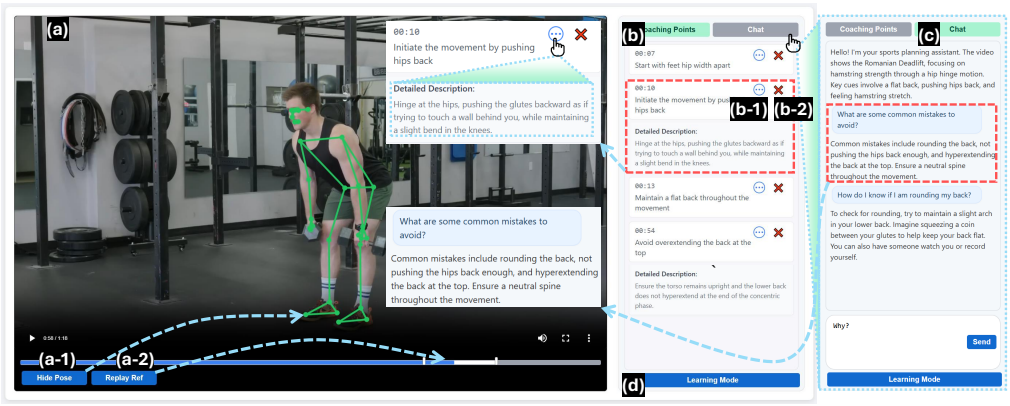

Figure 4: Learning Mode user interface. (a) Reference video panel. (a-1) Toggle an overlaid pose skeleton for the coach; (a-2) loop the selected segment for focused repetition. (b) Coaching points panel. (b-1) Selecting a point opens the cue with an expandable detailed explanation; (b-2) remove a point to tailor the plan. (c) A free-form, LLM-powered (Gemini) chat lets users ask clarifying questions (e.g., common mistakes, self-checks) and returns cue-aligned answers. (d) Advance to *Practice Mode* via the footer action.

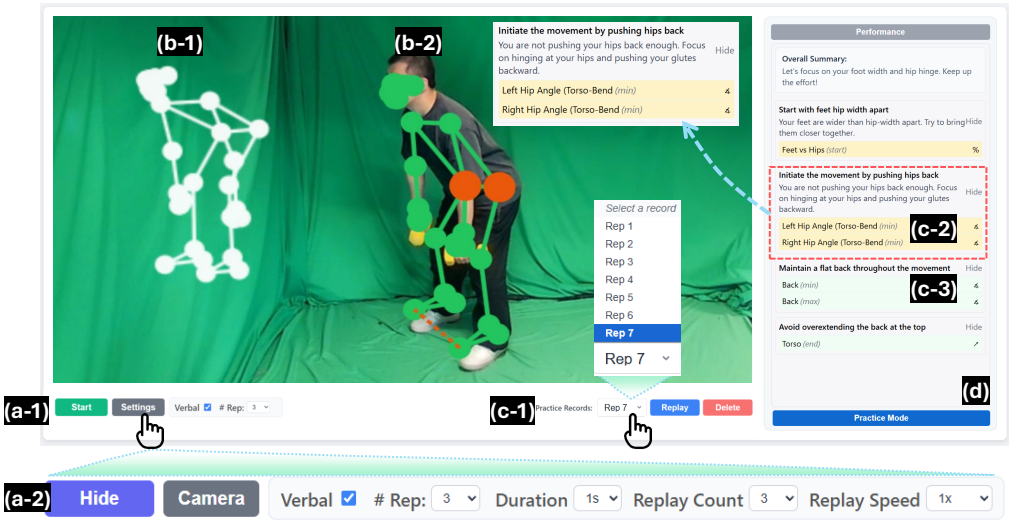

Figure 5: Practice Mode user interface. (a-1) Session controls: press *Start* to begin a multi-trial session and open *Settings* to configure options; (a-2) quick toggles for capture and replay (e.g., camera, verbal feedback, # repetitions, duration, replay count, replay speed). After each trial, the system replays the current attempt with overlays of the reference skeleton (b-1) and the user's skeleton (b-2), highlighting correct regions in green and incorrect regions in orange. After the session, users can select any trial from the record list (c-1) for replay, while the *Performance* panel presents an overall summary and CP-wise outcomes for each parameter (pass in green c-2, fail in orange c-3). Users can switch back to Learning Mode via (d).

the user optionally adjusts the key reference segment to calibrate evaluator targets (time window and pose anchors). They proceed to practice via the footer action (Figure 4 d).

Upon entering *Practice Mode*, the user sets session options such as number of repetitions, capture/replay toggles, enable/disable Text-to-Speech (TTS) verbal feedback, and replay speed (Figure 5 a-1,2). Pressing *Start* begins a multi-trial session. For each repetition, the system (i) records the attempt with a 3-second countdown, (ii) runs CP-wise evaluators in real time, and (iii) auto-replays the attempt alongside the reference skeleton (Figure 5 b-1). Overlays highlight correct regions in green and incorrect regions in orange (Figure 5 b-2), while concise CP-wise verbal feedback is delivered via TTS. After the session, users can select any trial from the record list for replay (Figure 5 c-1). A *Performance* panel presents an overall summary and CP-wise outcomes (pass/fail) with parameter-level details (Figure 5 c-2,3). Users can return to *Learning Mode* (Figure 5 d) to refine CP selections or revisit explanations, then iterate the practice–review loop.

5.2 Knowledge-Base Construction

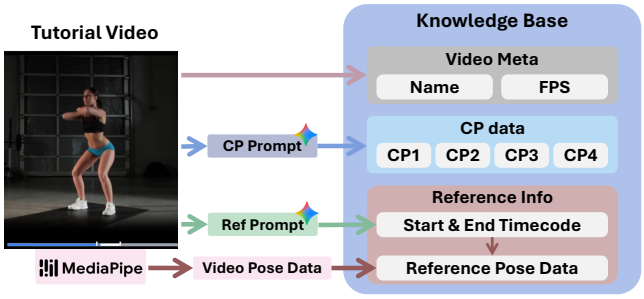

Figure 6: Knowledge base construction pipeline. We extract video metadata, use Gemini to obtain coaching points and reference timecodes with two different prompts, and run MediaPipe to generate pose data. The pose sequence is then sliced using the reference segment, producing a knowledge base containing video metadata, CPs, and reference pose info.

As shown in Figure 6, the first stage has two main tasks: (1) extracting coaching points from the tutorial video; and (2) identifying a segment that contains exactly one repetition of the skill. We invoke Gemini twice under the **Analyzer** agent, using two different prompts. With a *CP-Extraction prompt* (P_{CP}), Gemini performs temporal segmentation and extracts posture-related CPs; each CP is represented by its name, a detailed description, and reference timecodes. With a *Reference-Localization prompt* (P_{REF}), Gemini localizes a *reference segment* from the video that contains a single complete repetition, with start and end timecodes. The system then runs pose estimation [48] on the entire video, the start and end timecodes are used to slice the corresponding pose sequence as the reference motion. The CP information and the reference motion are stored in a structured Knowledge Base (**KB**) JSON format (Figure 7). This representation enables downstream components to access detailed movement criteria for evaluation and feedback generation.

5.3 CP-Wise Evaluator Compilation

The second stage of our workflow is managed by the **Builder** agent. Given the semantic CPs and the reference motion S^{ref} in the **KB**, Builder turns text-level CPs into executable evaluators and initializes their reference targets. As shown in Figure 7, the stage proceeds in two steps: (i) CP-to-parameter mapping to enable per-CP evaluation, (ii) parameter-to-evaluator initialization with reference motion.

CP-to-Parameter Mapping. While CPs provide a semantically meaningful description of movement intent, they remain abstract and cannot be directly measured from raw motion data. To enable computational evaluation, **Builder** first maps each CP onto a set of measurable parameters. Formally, we represent this mapping as:

$$CP_i \;\rightarrow\; \{\, P_k = (T, J, E) \;\mid\; k = 1, \ldots, n \,\}$$

where CP_i denotes the i-th coaching point, and each P_k corresponds to a measurable parameter defined by T as the parameter type, J as

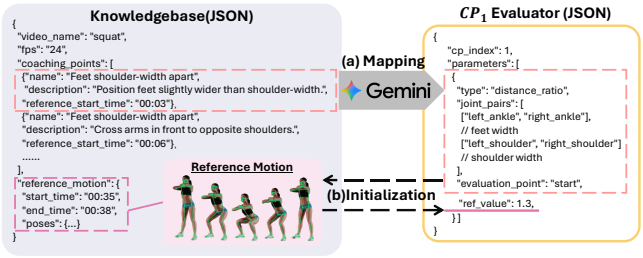

Figure 7: CP evaluator initialization. (a) Each coaching point (CP) from the KB is mapped to measurable parameters (T, J, E) which can be compiled into callable Python evaluators. (b) The evaluators are applied to the reference motion to compute per-parameter reference values (`ref_value`), which are stored in a CP-evaluator JSON.

the relevant joints, and E as the temporal evaluation point. **Builder** transfers each CP description against a predefined *Parameter Library* to generate a structured mapping. With a *Mapping prompt* (P_{MAP}), we invoke Gemini for this step and augment the prompt with the parameter-library schema and examples, yielding the per-CP mapping (Figure 7 a).

Parameter-to-Evaluator Compilation. Given the mapping result for each CP_i as $P_k = (T, J, E)$, the **Builder** is able to compile each parameter into a callable evaluator

$$f_{i,k} \;\equiv\; \textsc{Compile}(P_{i,k}) : \; \mathcal{S} \rightarrow \mathbb{R},$$

where $\mathcal{S}$ is the space of pose sequences and $f_{i,k}(S)$ returns a scalar. Applying the evaluators to the reference motion S^{ref} yields reference targets $v^{ref}_{i,k} := f_{i,k}(S^{ref})$. We store, per CP, the pairs $\left(f_{i,k},\, v^{ref}_{i,k}\right)$ as the **CP-wise evaluator** set (Figure 7 b).

$$\mathcal{F}^{ref}_i = \left\{\, \left(f_{i,k},\, v^{ref}_{i,k}\right) \right\}_{k=1}^{n_i}.$$

For downstream use, each CP's evaluator is serialized as a compact JSON file. This set constitutes the quantitative benchmark against which user performance will be compared.

5.4 User Evaluation

With the reference evaluators $\{\mathcal{F}^{ref}_i\}$ established, the system assesses each user trial against the reference values. For each trial, the system captures a pose stream via a webcam. Dynamic Time Warping [62] aligns the pose stream to the reference motion S^{ref} and locates the executed single-repetition segment.

For a given CP i, the same evaluator set is applied to the localized user segment to obtain per-parameter user values:

$$v^{usr}_{i,k} := f_{i,k}(S^{usr}), \quad k = 1, \ldots, n.$$

Contrasting the user value with the corresponding reference yields deviation metrics:

$$e_{i,k} \;=\; v^{usr}_{i,k} - v^{ref}_{i,k}, \qquad |e_{i,k}| \;=\; \left|v^{usr}_{i,k} - v^{ref}_{i,k}\right|.$$

A parameter is marked *misaligned* if $|e_{i,k}| > \tau$ for a predefined per-parameter threshold τ based on the parameter type; otherwise

it is *aligned*. For convenience, define the binary indicator $z_{i,k} := \mathbf{1}\left(|e_{i,k}| > \tau_{i,k}\right) \in \{0, 1\}$. The per-trial CP-wise results are then

$$R^{(t)} = \{r_i^{(t)}\}_{i=1}^{N_{\text{CP}}}, \qquad r_i^{(t)} = \left\{(v_{i,k}^{usr}, v_{i,k}^{ref}, e_{i,k}, z_{i,k})\right\}_{k=1}^{n_i},$$

where n_i is the number of parameters associated with CP_i. These results are appended to a running history $\mathcal{H}$ used by the contextual-feedback generation.

5.5 Contextual Feedback Generation

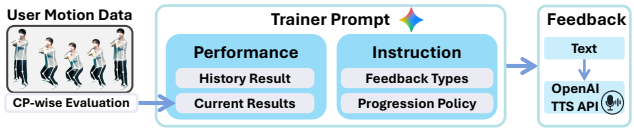

Figure 8: Verbal Feedback generation pipeline. For each user attempt, the system performs CP-wise evaluation and constructs a performance context from both current and historical results. Combined with instruction context (feedback types and progression policy), the Trainer generates adaptive verbal feedback, which is synthesized via the OpenAI TTS API.

The final stage of our workflow is handled by the **Trainer** agent as shown in Figure 8. It processes the stream of **Coaching Point-wise Results** and fuses them with a historical context to generate adaptive feedback.

History Context. For each CP, the system maintains a memory window capturing: (i) recent signed and absolute errors for parameters of all CP_i; (ii) recent feedback history for each CP_i and recent overall and verbal feedback text; (iii) an improvement label $L_i \in \{\text{improving}, \text{regressing}, \text{stable}\}$ computed by comparing the current absolute error with the windowed average; (iv) a misalignment frequency $V_i = \sum_{\ell=1}^{t} \sum_k z_{i,k}^{(\ell)}$ calculated from (i).

Feedback Types. We operationalize feedback as six atomic types (as mentioned in Section 3) that encode conversational intent and information granularity: 1. Standard Correction: first-time misalignment, concise corrective action. 2. Detailed Explanation: repeated misalignment, short why+how anchored to CP details and deviations. 3. Motivational Praise: just corrected a prior misalignment. 4. Confirmation: consistently aligned performance. 5. Brief Reminder: regression after praise/confirmation. 6. Silence: flawless performance or redundant repetition.

Progression Policy. Inspired by our formative study, the system assigns a feedback type to each CP using a history-aware progression: novel misalignments begin with type 1, persistent misalignments escalate to type 2 (drawing on CP descriptions and deviations), successful recovery advances to type 3, sustained alignment transitions to type 4 and gradually to type 6, and regressions after praise/confirmation trigger type 5.

Prompt Assembly and Output. Finally, all ingredients are synthesized into a generative prompt. The core instruction is defined by the *Feedback Types* and their *Progression Policy*, while the performance contextual inputs include: (i) the per-CP history context and (ii) the current CP-wise results. These elements are combined into a structured prompt and passed to the **Trainer** agent. The agent then generates the final verbal feedback text, which is converted into speech via a TTS engine for in-situ delivery. For visual feedback, the system uses the current CP-wise results to render key joints or limbs requiring correction are highlighted on the user's on-screen avatar, providing an intuitive visual guide.

5.6 Implementation Details

The system was developed and tested on a desktop NVIDIA GeForce RTX 2080 Ti GPU, which facilitated real-time model inference. A standard 1080p webcam operating at 30 FPS was used for video capture. The frontend UI was built with React, and the backend with Python. We used the MediaPipe with its "Heavy" model to extract 33 body keypoints. To optimize performance, the **Analyzer** and **Builder** agents used the Gemini-2.5-flash model, while the Trainer agent used Gemini-2.5-flash-lite to achieve 2s feedback latency. Misalignment thresholds (τ) were empirically set based on parameter type (e.g., 15 degrees for angles, 10% deviation for distances) based on pilot feedback. The contextual feedback mechanism maintained a history of the last 5 trials. The verbal feedback was delivered using OpenAI TTS API[2].

6 Preliminary System Validation

In this session, we validate the computational functionalities of intermediate system stages using annotated ground truth from Section 4.2, focusing on two components: (i) *CP extraction* and (ii) *CP-to-parameter mapping*.

6.1 CP extraction

To evaluate whether our system can, for each video, generate a CP list that textually covers the *Gold CP list*, we use the videos along with their golden CPs. The *Gold CP list* refers to the manually annotated CP set established in Section 4.2. An experiment was done with the following settings:

- **Baseline (zero-shot):** Gemini-2.5, zero-shot with a meta prompt.
- **AgentCoach (few-shot):** additionally inject examples to explicitly teach the model to include content and exclude non-actionable content.

Matching rule and metrics. For each video, Gemini-2.5 (temperature = 0) serves as a semantic judge. Following [65], given a predicted CP p_i and a Gold CP g_j, we ask it to output a binary equivalence label. We then build a bipartite graph between $P = \{p_i\}$ and $G = \{g_j\}$ with edges only where equivalence is true. To ensure one-to-one alignment, we compute a *maximum cardinality bipartite matching* M (allowing imperfect matches; unmatched predictions count as false positives and unmatched Gold CPs as false negatives).

From M we derive: $\text{TP} = |M|, \text{FP} = |P| - |M|, \text{FN} = |G| - |M|$. Then compute the precision, recall, and F1 evaluation metrics. We report both *macro* averages (averaging per-video metrics) and *micro* averages (summing TP/FP/FN across videos).

[2]https://platform.openai.com/docs/guides/text-to-speech

We used all 55 annotated tutorial videos from Section 4.2 for the CP extraction experiment. Table 1 reports macro- and micro-averaged scores. We observe that including the few-shot examples improves both precision and recall; therefore adopt the few-shot setup as the default for AgentCoach.

Table 1: CP extraction results. Macro- and micro-averaged Precision, Recall, and F1 are reported (higher is better).

Condition	Precision	Recall	F1
	Macro Avg.		
Baseline (zero-shot):	77.72	90.01	81.12
AgentCoach (few-shot)	**88.21**	**96.41**	**91.57**
	Micro Avg.		
Baseline (zero-shot):	72.40	89.16	79.91
AgentCoach (few-shot)	**84.78**	**96.06**	**90.07**

6.2 CP-to-parameter mapping

Goal. Evaluate whether the system, given a *video-level batch input* (the full CP list of a video plus meta information), correctly maps *each* CP to its unique parameter set $\{P_k\}$ as defined in the parameter library. A CP counts as correct only under *exact set equality*.

Ablation Settings.

- **Baseline (zero-shot):** Gemini-2.5 with zero-shot prompting using only the parameter-library schema.
- **AgentCoach (few-shot):** inject mapping examples to guide structured output.

Input/Output per video. For a video v, let the annotated Gold mapping be

$$\mathcal{G}_v = \{\, (CP_{v1}, G_{v1}), \ldots, (CP_{vN_v}, G_{vN_v}) \,\},$$
$$\text{where} \quad G_{vk} = \{\, (T, J, E)_\ell \mid \ell = 1, \ldots, n_{vk} \,\},$$

where each tuple (T, J, E) follows the taxonomy in Section 4.2. Given the batch input $\langle C_v$, video meta$\rangle$, the system returns the predicted mapping

$$\hat{\mathcal{P}}_v = \{\, (CP_{v1}, P_{v1}), \ldots, (CP_{vN_v}, P_{vN_v}) \,\},$$
$$\text{where} \quad P_{vk} = \{\, (T, J, E)_\ell \mid \ell = 1, \ldots, \hat{n}_{vk} \,\}.$$

Evaluation. We formalize both P_{vk} and G_{vk} (normalize joint aliases with Mediapipe index, sort, fix field order) and define

$$\text{hit}_{vk} = \begin{cases} 1, & \text{if } P_{vk} = G_{vk} \text{ (element-wise exact match)}, \\ 0, & \text{otherwise.} \end{cases}$$

Per-video exact match:

$$\text{EM}_v = \frac{1}{N_v} \sum_{k=1}^{N_v} \text{hit}_{vk},$$

and we report macro-EM (average over videos) and micro-EM (average over all CPs).

Results. We split the annotated CP from Section 4.2 into a small exemplar set and an evaluation set. The exemplar set contained 5 videos sampled to cover all four parameter types and all temporal points, which were injected as few-shot examples. Table 2 summarizes the performance under batch inference. We observe that adding the few-shot exemplars improves both macro- and micro-EM.

Table 2: CP-to-parameter mapping with video-batch inference (higher is better)

Condition	EM (macro)	EM (micro)
Baseline (zero-shot)	87.68	86.27
AgentCoach (few-shot)	**93.69**	**94.06**

7 User Study

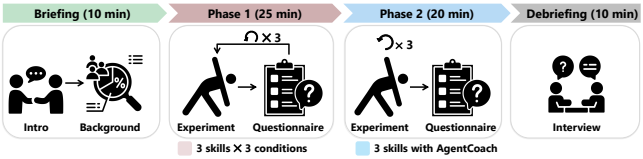

Figure 9: Overview of user study procedure with four parts: briefing, phase 1, phase 2, and debriefing.

We conducted a controlled, within-subjects study in two phases to evaluate AgentCoach against two other conditions and to assess feature-level and overall experience (Figure 9). Each participant practiced a total of 6 motor skills (Appendix C): three in Phase 1, each paired with one of the three feedback conditions; and three additional skills in Phase 2 using the full AgentCoach system. The study lasted 60–70 minutes, including a briefing and a debriefing, and was approved by the institutional review board (IRB).

7.1 Participants and Apparatus

We recruited 24 adults from the university and the local community (13 male, 11 female; $Mean_{\text{age}}$=22.50, range 18–40). All participants were able to perform physical activity and follow movement instructions in English. We excluded individuals who reported a current musculoskeletal injury, recent surgery, or other contraindications to light exercise. Participants received a $20 gift card upon completion of the study. The study was conducted in the office with a 63-inch 4K TV positioned approximately 3.0 m from the participant to present the reference video, feedback, and dashboards. A front-facing 1080p webcam clipped to the top edge of the TV captured full-body motion at 30 fps.

7.2 Conditions

We compared three feedback conditions - C1, C2, and C3 - that differed in modality and spoken-guidance adaptivity. The conditions are detailed below.

- **C1—Visual Highlighting** provides visual-only feedback: a reference-pose skeleton showing the target motion/pose, and a user-pose skeleton with misaligned joints or limb segments highlighted. *No audio.*

Figure 10: User study exercises for Phase 1 (a), Phase 2 Set A (b-1), and Phase 2 Set B (b-2).

- **C2—Data Diagnostics** extends C1 with diagnostic spoken narration (similar to AIFit [24]) that enumerates which joints or limb segments deviate from the reference. This condition is descriptive—reporting what diverged—without prescriptive advice or any history-based adaptation.
- **C3—AgentCoach** extends C2 with performance-history-aware, CP-based verbal feedback that provides concise prescriptive suggestions for the next attempt. In addition, a CP performance panel shows pass/fail per CP alongside the C2 diagnostics.

7.3 Procedure

7.3.1 Briefing (10 min). Participants completed a consent form and a demographic/background questionnaire. They then received a short tutorial on AgentCoach: how to start and end a session, how feedback appears, how to switch modes, and how to play the reference video and use the checkpoint. Each participant performed 2–3 practice trials with a warm-up movement to confirm system framing and comprehension. Both Phase 1 and Phase 2 were conducted within the learning–practice workflow. We detail the steps for each mode below.

Learning mode. For each skill, participants watched the instructional video and could replay the reference segment. The system then presented the CPs extracted from the video. Participants reviewed these CPs, asked clarification questions via the chatbox if needed, and then proceeded to the practice mode.

Practice mode. In this mode, participants performed the selected skill followed by clicking the *Start* button. We set each skill replication to consist of two sessions of five repetitions (10 trials total). Within each session, repetitions were continuous, and after each repetition, the system delivered feedback based on the assigned condition. Participants were free to return to the learning mode between sessions or to consult the chatbox.

7.3.2 Phase 1 (25 min). Each participant experienced three feedback conditions, each paired with a different skill. The order of the three skills was fixed, but the assignment of conditions (C1–C3) to skills was counterbalanced across participants, yielding 3! = 6 permutations, so the condition order varied by participant. The three Phase 1 skills were (Figure 10 a): Dumbbells (Romanian deadlift), Yoga (Trikonasana pose), and Pickleball (forehand drive). After completing each skill, participants filled out a post-condition questionnaire about the feedback they had just received.

7.3.3 Phase 2 (20 min). To assess generalizability and usability, participants practiced six additional motor skills with **C3—AgentCoach**. These were divided into two sets: Set A (Figure 10 b-1) included Soccer (passing), Table tennis (forehand drive), and Single-leg hip thrust; Set B (Figure 10 b-2) included Breakdance (cross step), Tennis (forehand drive), and Squat. Participants were evenly assigned to the two sets. Each set covered a leg-dominant movement, a racket sport, and a general exercise, ensuring diverse movement types. This phase evaluated how well the system's feedback supported different categories of motor skills and its overall usability.

7.3.4 Debriefing (10 min). Participants then completed a *post-study questionnaire* and an *end-of-study survey* including open-ended questions on preferred features, appropriate use contexts, potential improvements, and concerns about the system.

7.4 Phase 1 Results

We aggregated ratings by condition (collapsed across skills) and computed per-participant means for each condition.

7.4.1 Questionnaire Results. We focused on six user–perceived properties of instructional feedback on 5-point Likert scales (1=Strongly Disagree, 5=Strongly Agree): *Comprehensibility* ("The feedback was clear and easy to understand."), *Actionability* ("The feedback gave me a concrete next step."), *Terminology* ("The wording aligns with sport/exercise terms I already know."), *Confidence* ("The feedback increased my confidence in my ability to perform the movement correctly."), *Recall* ("I can still recall the key coaching points after the session."), and *Mental Demand* ("Processing the feedback felt mentally demanding."). Our items were adapted from prior work on formative feedback [57] and align with constructs frequently used in HCI evaluations [86]. We ran within-subject Wilcoxon signed-rank tests for pairwise contrasts, and reported per-condition summary statistics across ratings below.

From understanding to acting. *Comprehensibility* improved from C1 (AVG = 2.38, SD = 1.18) to C2 (AVG = 3.12, SD = 1.01) and C3 (AVG = 4.12, SD = 0.88). *Actionability* showed the same progression—C1 (AVG = 2.62, SD = 1.07), C2 (AVG = 3.00, SD = 1.12), C3 (AVG = 4.21, SD = 0.87). Across both ratings, means increased from C1 to C3 while standard deviations generally decreased. In pairwise contrasts, C3 differed significantly from both C1 and C2 (all $p<.001$). The largest gain in average score was from C2 to C3 (+1.09 for Comprehensibility; +1.21 for Actionability), indicating that the feedback became increasingly execution-ready. Participants put it plainly: *"I didn't have to interpret it—I just did it."* (P12); *"It read like a small checklist I could follow."* (P7)

Same Training Vocabulary. *Terminology* increased from C1 (AVG = 2.96, SD = 1.24) and C2 (AVG = 3.21, SD = 1.12) to C3 (AVG = 3.83, SD = 0.85) (both $p<.01$). One participant noted: *"The terms matched what my coach would say."* (P5) These results suggest that participants perceived clearer terminology, a more coherent structure, and more specific next steps.

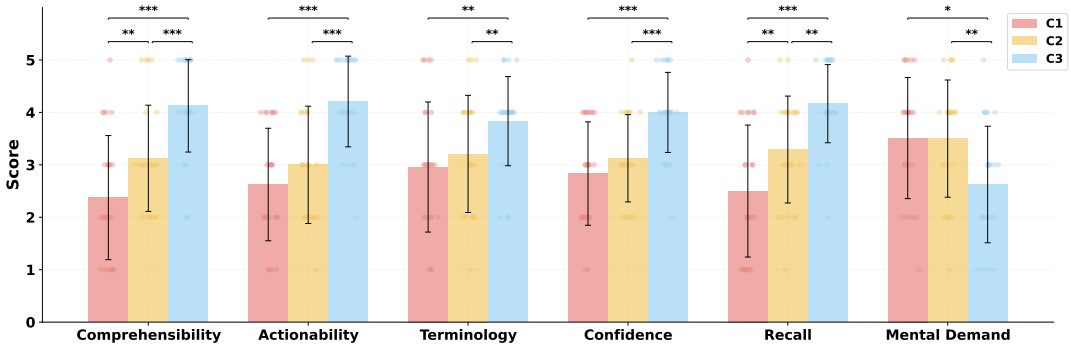

Figure 11: Phase 1 results. Brackets show pairwise Wilcoxon signed-rank tests ($^{*}p<.05$, $^{**}p<.01$, $^{***}p<.001$). Higher values indicate more positive ratings.

Higher control and memory. *Confidence* was higher in C3 (AVG = 4.00, SD = 0.76) than in C2 (AVG = 3.19, SD = 0.85) and C1 (AVG = 2.95, SD = 0.95) (both $p<.05$). *"Corrections felt specific enough to commit to."* (P3) *Recall* was likewise higher in C3 (AVG = 4.14, SD = 0.77) than in C2 (AVG = 3.43, SD = 1.00) and C1 (AVG = 2.67, SD = 1.25) (both $p<.05$). *"I could still list the key points afterward."* (P9) Taken together, these results suggest that richer feedback did not just persuade; it also helped users improve self-checking and retain actionable cues.

Lower mental effort. *Mental Demand* was lower in C3 (AVG = 2.62, SD = 1.11) than in C1 (AVG = 3.51, SD = 1.15) and C2 (AVG = 3.50, SD = 1.12) ($p<.05$,vs. C1; $p<.01$,vs. C2), consistent with participants' reports of spending less time "processing" the message.

Summary. Across the first four ratings, the average score increase from C2 to C3 exceeded the increase from C1 to C2, highlighting the value of CP-wise, prescriptive feedback over diagnostic feedback. From visual-only feedback (C1) to visual & diagnostic spoken feedback (C2) to visual & prescriptive, adaptive spoken feedback (C3), the first five ratings increased monotonically, *Mental Demand* decreased, and standard deviations were generally smaller in C3. Overall, C3 yielded feedback that participants perceived as clearer, more executable, better remembered, and less effortful to process. These patterns suggest a design implication: prioritize structured guidance that affords immediate action and aligns with users' vocabulary, rather than adding diagnostic text alone.

7.4.2 Learning Curves by Condition. Building on Phase 1—where richer feedback improved perceived clarity and actionability—we examined whether those gains translated into objective performance across repetitions (Figure 12). We analyzed ten repetitions per participant (two sessions of five). For each participant and repetition, *accuracy* was the fraction of coaching points passed in that repetition; we then averaged accuracy across participants for each repetition. The lines show the mean accuracy per-repetition, and the shaded bands denote the standard deviation across participants.

Session 1 (R1–R5). Conditions accuracy started at similar levels (R1 ≈ 0.83–0.86). All three conditions improved over the first five repetitions; C3 increased most smoothly, with variability narrowing by R4–R5; C1 improved more erratically, and C2 fell in between.

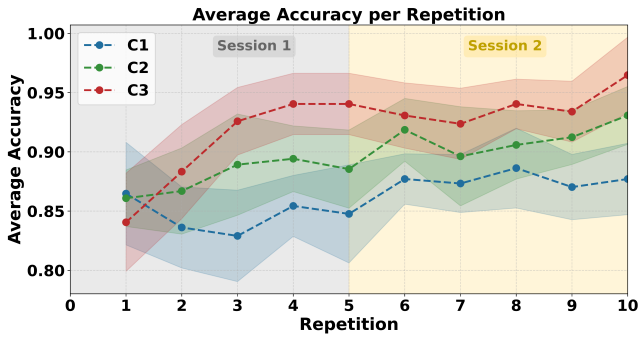

Figure 12: Averaged Learning Curves for C1–C3. Background shading separates two sessions (Session 1: reps 1–5; Session 2: reps 6–10).

Between sessions intervention. After R5, participants could replay their own videos (and, in C3, review per-CP results). At R6, both C1 and C2 showed a marked step change; C2 exhibited the larger jump, whereas C3 was flat or slightly lower before resuming its upward trend.

Session 2 (R6–R10). Except for the step at R6, C2 and C3 increased gradually with similar slopes. C1 largely plateaued and fluctuated around a flat mean. By R10, C3 achieved the highest mean accuracy (≥0.95) and the greatest overall increase from R1 to R10; C2 ranked second (≈0.93); C1 remained the lowest (<0.89). In later repetitions, variability under C3 was lower than under C1.

Across both sessions, C3 consistently produced the strongest learning signal—namely, higher end performance and greater total improvement—relative to C1 and C2. The post-break jump at R6 suggests that targeted review helps consolidate learning; however, C3 sustained gains thereafter without the same level of reliance on chat seen in C1 and C2, indicating that timely, in-task guidance supports steady improvement.

7.5 Phase 2 Results

We assessed two aspects of experience on a 5-point Likert scale. *Feature Evaluation* asked whether the core learning mode features (Q1–Q3) and practice mode features (Q4–Q6) helped users plan and

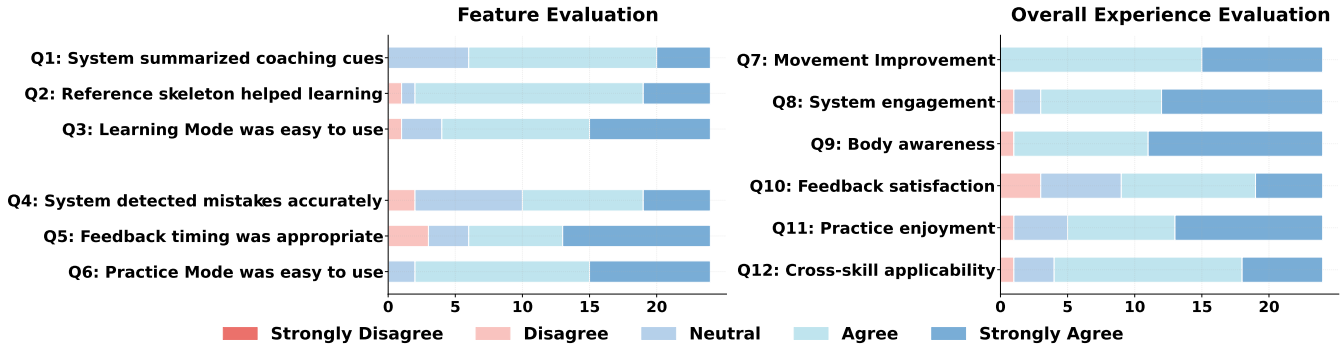

Figure 13: Post-study ratings. Left: Feature Evaluation for Learning Mode (Q1-Q3), and Practice Mode (Q4-Q6); Right: Overall Experience Evaluation.

make in-action corrections. *Overall Experience Evaluation* assessed perceived improvement, engagement, awareness, satisfaction, enjoyment, and cross-skill applicability (Q7–Q12).

7.5.1 Feature evaluation. Participants started by replicating additional motor skills and then tested whether the system could correct them in action. They reported that the system could successfully pull coaching points into a workable summary Q1 (AVG = 3.92, SD = 0.65) and made the target motion concrete via the reference skeleton Q2 (AVG = 4.08, SD = 0.65). One typical reaction was, *"A quick glance at the skeleton gives me a first focus—hips back before lift—so I know what to try next."* (P16). Learning mode was easy to use Q3 (AVG = 4.17, SD = 0.82) and remained smooth during practice. The AgentCoach system automatically proposes the start and end timestamps for the reference clip. 15 of 24 participants confirmed the system-generated clip as the reference without adjusting either marker for any of the videos. The remaining participants made minor adjustments, refining on average 1.33 of the 3 clips used in Phase 2. These timestamp edits typically took under 10 seconds.

In practice, participants judged the timing appropriate Q5 (AVG = 4.08, SD = 1.06), consistent with a 2–3 s end-to-end latency from motion to cue in our pipeline. As one participant put it, *"on time, with an occasional brief lag when the text cue was generated."* The practice mode was easy to operate Q6 (AVG = 4.29, SD = 0.62). Perceived mistake detection was moderate Q4 (AVG = 3.71, SD = 0.91). In our deployment, MediaPipe's pose estimates degraded with loose clothing and self-occlusion, "*sometimes the detection didn't feel accurate.*" (P2).

7.5.2 Overall User Experience. For the overall experience, the average ratings were in the *Agree* range: five of six items were larger than 4.0. First, participants credited the system with supporting improvement during practice sets Q7 (AVG = 4.33, SD = 0.48)—a strong signal that the CP–to–feedback loop was working, even as detection remained at a moderate level. They also described the sessions as engaging Q8 (AVG = 4.33, SD = 0.82) and, notably, as heightening what they notice about their own bodies Q9 (AVG = 4.46, SD = 0.72), the highest system score. One participant summarized the change as, *"I now catch my posture drifting even when I'm not using the app."* Enjoyment was also high Q11 (AVG = 4.21, SD = 0.88), which likely supports return use. Feedback satisfaction was moderate Q10 (AVG = 3.71, SD = 0.96). Follow-up comments indicated that dissatisfaction mainly reflected detection inaccuracy rather than the feedback format: *"Sometimes the detection wasn't accurate; the feedback itself was fine."* (P13). By contrast, cross-skill applicability was positive Q12 (AVG = 4.04, SD = 0.75), suggesting the cue grammar transfers well across different sports skills. Meanwhile, we also obtained a mean SUS score of 75.42 for the UI, which falls between the adjective ratings "Good" and "Excellent" [6].

In general, users liked the system; when the next move was made concrete in time and space—via the skeleton (Q2), appropriate timing (Q5), and an easy-to-use Practice interface (Q6)—participants reported improvement (Q7) and, at the system level, higher engagement (Q8), awareness (Q9), and enjoyment (Q11).

7.5.3 Communication with the chatbox. We analyzed all chat messages addressed to the chatbox across the study. In total, the 24 participants asked 33 questions (mean > 1 per person). Four participants asked none; one participant asked 4 questions; three asked 3; four asked 2; and the remaining twelve asked 1. When aligned by condition, 45.5% of questions occurred after C1, 39.4% after C2, and only 15.2% after C3. This suggests that with C3, participants had fewer clarification needs, consistent with our observation of reduced chat usage under this condition. Appendix D shows the word cloud of all questions from participants. The most frequent terms include body parts (e.g., *hip, leg, foot, torso, shoulder*), laterality (*left/right*), and geometric cues (*angle, line, range*). Overall, participants mainly sought confirmation about segment alignment, side-specific cues, and acceptable ranges of motion, rather than open-ended or off-task queries.

8 Discussion

8.1 Verbal feedback enhances visual understanding

Visual cues make spatial misalignment immediately apparent. Error highlights can be "where-focused" rather than "how-to"—useful for locating an error but sometimes insufficient for specifying the next action. As one participant put it, *"I could see the problem, but I still needed a do-this-now instruction."* (P20). Moreover, watching visual cues on the screen requires users to keep their head and gaze fixed, which is not suitable for all cases. Another user mentioned, *"I have*

to keep my eyes skyward, so I'm not able to see my performance on the screen." Previous work [37, 96] tried to have the video follow the user's view within a virtual-reality head-mounted display (HMD); however, HMDs are still not accessible for most people.

Verbal cues were often favored when visuals alone were not sufficiently interpretable or when users wanted an explicit guide that converts a highlight into an action (e.g., "rotate your hips more") that links to a coaching point. Besides, verbal cues are also more accessible in eyes-busy, hands-occupied settings and embed readily into portable devices (e.g., earbuds, smartwatches), enabling heads-up use and reducing on-screen text load. This pattern is consistent with the modality principle from multimedia learning: pairing graphics with brief spoken words generally outperforms graphics plus on-screen text because it splits processing across auditory and visual channels and reduces split attention [54]. In our experiment, both verbal conditions improve learners' understanding of the feedback (Section 7.4). *"The verbal feedback tells me whether it's a joint-angle or limb-orientation issue; that's more direct than visuals alone."* (P2). *"From the visuals I could tell my shoulder line was off, but I didn't know what to do; the verbal cue 'rotate your shoulder' made it immediately clear."* (P3). Taken together, verbal acts as a thin "translation layer" that turns spatial markings into do-now actions; when delivered as brief, bandwidth-triggered prescriptions, they increase actionability without inflating cognitive load [66, 68].

8.2 CP-based Feedback Enhances Learning Experience

CP-based, history-aware feedback (C3) adds three ingredients beyond diagnostics (C2): *selection* (which CPs matter now), *prescription* (what to do next), and *progress memory* (how this CP's performance has evolved). In practice, this turned a list of deviations into a prioritized, bite-sized plan for the next repetition.

First, CP prioritization reduced cognitive cost. Rather than listening to a metrics report, participants said the system's CP suggestions made the next attempt obvious: *"Don't read me all the numbers—just say 'rotate hips.' With that, I know exactly what to try next."* (P5). This was especially salient in situations where multiple misalignments occurred simultaneously.

Second, prescriptions phrased at the CP level improved perceived usefulness and trust because they linked actionable cues to the tutorial content. *"When it says 'push your hips backward more' and highlights the hip joints, it clicks. The cue matches what the coach said in the video, so I trust it."* (P12).

Third, the history-aware progression (escalate on repeated misalignment; praise on recovery; remind on regression) provided balanced guidance and autonomy. Participants described the cadence as "present when needed, quiet when stable," aligning with guidance-hypothesis recommendations to fade concurrent guidance as proficiency stabilizes: *"Early on, it nudged me to bend my knees; once I got it, it mostly confirmed or stayed quiet. It felt like it was paying attention to my progress instead of repeating itself."* (P16).

8.3 Tone and Coaching Style

Participants diverged in how they reacted to "strict" feedback when they repeatedly made the same mistake. One participant noted, "*If it talks to me like that again, I'll stop using it*" (P7), adding, "*People can be emotional, but a machine 'having emotions' just makes me unhappy*" (P7). Another participant downplayed the issue: "*It was fine for me*" (P3). We did not intentionally design a harsh persona; the few "strict" episodes arose from prosody spikes in TTS and occasional LLM phrasing. These were rare but salient. To interpret participants' reactions, we draw on work showing that agent legibility/predictable behavior and alignment with users' social expectations shape acceptance of agent feedback [8, 31].

Social Presence and Intent Clarity Shape Tone Perception. Two factors likely explain the split reactions. First, identical content is experienced differently under higher social presence: when feedback sounds more "human-like," tone becomes socially consequential and is policed by interpersonal norms [8]. Second, *expectation alignment* matters: when firmness appears without an explicit, user-endorsed rationale, some users read it as an *expectation mismatch*. By *predictability*, we mean a stable mapping from similar error contexts to similar tone; by *transparency*, we mean brief "why-now" explanations and visible user controls for tone. Both reduce surprise and support trust [31]. This framing helps explain why a brief, sudden loudness or intonation felt unacceptable to one participant but inconsequential to another.

Coach Perspective: Fit Style to Age and Purpose. A coach with 5 years' experience emphasized that tone depends on learner age and goal: "*With kids I never use harsh words—only encouragement. For competition training, we are strict; every detail must be right. For social or fitness use, tolerance is wider*" (P7). This aligns with coaching guidance to match strictness and prioritization to the learner's stage and objective—stabilizing prerequisites before tightening criteria, and avoiding one-size-fits-all escalation [55, 68]. While this points to promising profile-conditioned policies (age/goal), our work targets broad, beginner-level use and does not evaluate age- or goal-specific tone strategies; future iterations can incorporate lightweight profiles and test differential effects across learner groups.

8.4 Toward Human Coaching Feedback

Our user study raised questions about how learners compare LLM-based feedback with human coaching. To probe this comparison more directly, we conducted semi-structured follow-up interviews (around 10 minutes each) with users who learned with coaches to understand how AgentCoach aligns with or differs from human coaching across quality, nuance, and responsiveness. All interviews were audio-recorded and transcribed for analysis. The interview questions are listed in Appendix E.

Quality. Participants found the system reliable for identifying technical errors, such as joint angles and stance. A strength trainee (P6, 5 years) notes that it "catches my round back." However, learners emphasized that human coaches also adjust expectations based on other contexts (e.g., body characteristics, learner level, as discussed in Section 8.6.4), which the system does not yet model. This gap suggests that AgentCoach could integrate contextual modeling in the future to approximate human decision-making better.

Nuance. Users felt that AgentCoach's verbal cues were clear and actionable, providing not only guidance but also confirmation. A

yoga learner (P2, 1 year) mentioned that the feedback is "concise but specific enough to try," especially when paired with visual highlights. A table tennis learner (P16, 2 years) also pointed out that human coaches integrate kinesthetic cues, such as muscle engagement and weight distribution, which video alone cannot capture. As noted in our limitations, the system cannot access these internal cues, though participants suggested wearable sensing as a future extension.

Responsiveness. Participants suggested that AgentCoach gives targeted, CP-dependent feedback and highlighted that, unlike human coaches, the system does not "get distracted" or miss recurring errors (P10, 3 years of tennis learning experience). Currently, AgentCoach delivers feedback after each trial due to the processing time required. Human coaches, however, vary in when they give feedback, for example, saying "heels down" the moment a squat starts to tilt, or "rotate now" as a tennis player begins the swing. Participants reported that AgentCoach cannot yet match this fine-grained timing. It could be mitigated in future work by reducing end-to-end latency to provide more immediate cues.

8.5 Comparison with Commercial Products

Recent years have seen a wave of commercial and AI-enabled fitness and rehab systems aimed at supporting users without human coaches. For example, Gymscore[3] offers computer-vision-based form analysis for strength and fitness training; Sportsbox AI[4] uses video to reconstruct users' movements in 3D and delivers biomechanical insights and corrective feedback for golf; Kemtai and Motion Coach[5] offer real-time, camera-based posture tracking with automated corrective cues for rehabilitation. These tools demonstrate the practical value of AI-driven motion learning systems. We differ from these applications in two key ways. First, existing commercial products focus on single-shot, corrective feedback rather than progress-aware coaching. By leveraging LLMs, our framework tracks CP-wise performance over time and adapts the type and tone of feedback based on users' progress. Second, AgentCoach grounds feedback semantically in tutorial videos by extracting coaching points and linking them to measurable parameters. By tracking CP-wise results over time and using LLMs to generate context-aware feedback, the system provides guidance that adapts to users' evolving performance rather than relying on a single holistic score. This shift moves from solely corrective feedback to semantic, progress-aware coaching.

8.6 Limitations and Future Work

While our study explores AgentCoach's capability to support diverse sports and exercises using online tutorial videos, several limitations remain.

8.6.1 Non-Postural Key Points. Our system focuses on postural patterns that can be parameterized by joints or limbs. However, many coaching points extend beyond posture and remain difficult to quantify. As discussed in Section 4.2, these include muscle activation (e.g., "engage your core"), weight distribution (e.g., "shift weight to your back leg"), and object-related factors such as racket angle. We envision future work that extends beyond camera-based sensing by integrating additional modalities—such as EMG [71] or EIT [99] to capture muscle activity, on-body devices to mitigate occlusion [92–94], and IMUs to track object orientation [78]—and by pairing linguistic feedback with richer tactile technologies [10, 41, 58] to deliver a more embodied, multisensory coaching experience.

8.6.2 Reference Video Limitation. Our framework requires a clean reference segment containing a single, complete repetition of the skill from a fixed viewpoint. During parameter library construction, we observed that some tutorial videos either lack such segments or interleave them with camera viewpoint changes. Although we rely on Gemini to coarsely identify reference segments, the results still require user refinement (as discussed in Section 7.5.1). Novice users may find it difficult to determine what constitutes a full repetition, introducing further uncertainty. Moreover, vague instructions (e.g., "loosen your wrist") remain difficult to operationalize. Lastly, our CP extraction relies on explicit instructional content, such as verbal explanations from the instructor or textual annotations like subtitles. We consider this requirement reasonable for a suitable tutorial video, as without such cues, the video will be purely demonstrative. In this study, we assumed that users could manually select videos to minimize these issues. For videos beyond our current scope, we envision future works to explore algorithmic improvements, such as pose reconstruction from moving cameras [70, 81], video-language video grounding [43] for better reference clip segmentation results, and fine-tuning LLMs for better interpretation of coaching language.

8.6.3 Scope of Skills. AgentCoach is best suited for discrete skills that can be segmented into single, repetitive units (e.g., squats, push-ups, table tennis strokes). However, it does not generalize well to long-form composite routines (e.g., dance choreography) or continuous sequential training protocols (e.g., HIIT circuits). These skill types involve non-repetitive sequences that go beyond the unit of analysis supported by our current CP-to-parameter mapping. One potential workaround is to decompose them into single-skill units and apply our pipeline individually. Future research could investigate extending CP-to-parameter mapping to capture higher-level temporal structures across composite or sequential skills.

8.6.4 Thresholds. Consistent with prior work [12, 49], our implementation uses fixed thresholds (τ), which were based on expert feedback. While suitable for a beginner-level setup, τ is not fixed in real coaching practice. Coaches adjust them based on various contextual factors, such as the learner's skill level, physical condition, body shape, training goals, and other relevant factors. In yoga and general fitness, instructors loosen thresholds for beginners or older adults to ensure safety, while tightening them for experienced practitioners who can pursue finer precision. In sports like table tennis, coaches allow broad correctness for novices but narrow the acceptable range for intermediate or advanced players (e.g., requiring more precise racket angles and stable swing trajectories). These examples show that thresholds are context-sensitive. They are not mathematically optimal values but practical judgments about what is safe, achievable, and instructionally meaningful for each learner. In our current implementation, τ is set once based

[3] https://www.gymscore.ai/
[4] https://www.sportsbox.ai/
[5] https://kaiahealth.com/motion-coach/

on expert feedback and kept fixed during deployment, avoiding ongoing coach/expert involvement. However, a fixed τ may not fully generalize across diverse body types and conditions. Future systems could consider treating τ as a dynamic parameter that is adapted based on coaching intent [77] and learner characteristics, rather than a fixed value.

8.6.5 Impact of Pose Tracking Errors. Our system runs on minimal hardware, requiring only a local PC with a webcam and a speaker. This setup makes it highly accessible and enables potential deployment on edge devices like mobile phones and tablets. However, the results of vision-based pose estimation, especially for real-time prediction, are not always stable and may produce incorrect detections. Although MediaPipe performs robustly without a green background (see examples in Appendix F), it is optimized for single-person tracking. When additional people appear, the model may lose the primary user tracking. Such failures can propagate downstream, causing incorrect or missing CP-wise measurements and degraded feedback.

9 Conclusion

This paper introduces AgentCoach, a LLM-powered system designed to transform motor skill learning by providing adaptive and personalized coaching feedback. By extracting key coaching points from instructional videos and mapping them to measurable kinematic parameters, AgentCoach effectively bridges the gap between high-level semantic guidance and low-level postural analysis.

Our user studies demonstrate the potential benefit of AgentCoach's adaptive, CP-wise feedback over visual-only and generic verbal feedback. The findings highlight that by delivering targeted, history-aware, and prescriptive advice, AgentCoach enhances skill acquisition, boosts user confidence, and reduces the mental effort required to process feedback. This approach not only makes the feedback more actionable but also fosters a more engaging and effective learning experience.

The implementation and evaluation of AgentCoach underscore the immense potential of leveraging large language models to create more human-like and effective automated coaching systems. While the current system focuses on postural and limb-based feedback for discrete motor skills, future work will aim to broaden its capabilities. This includes incorporating a wider range of feedback modalities to address non-postural cues such as muscle engagement and weight distribution, as well as extending the framework to support more complex, sequential, and continuous motor skills.

In essence, AgentCoach represents a significant step forward in the development of intelligent systems for motor skill training. Its ability to provide nuanced, adaptive, and context-aware feedback paves the way for more accessible and personalized coaching, ultimately empowering individuals to learn new skills more efficiently and effectively.

Acknowledgments

We wish to thank all the reviewers for their invaluable feedback. This work is partially supported by the NSF under the Future of Work at the Human-Technology Frontier (FW-HTF) 1839971 and NSF Partnerships for Innovation Technology Transfer (PFI-TT) 2329804. We also acknowledge the Feddersen Distinguished Professorship Funds. Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency. Additionally, we thank the coaches and athletes we interviewed for their insights and suggestions that informed this work, with special thanks to Nishant Vasan and Ziyi Li.

References

[1] Manfred Agethen and Daniel Krause. 2016. Effects of bandwidth feedback on the automatization of an arm movement sequence. *Human Movement Science* 45 (2016), 71–83.

[2] Faisal Ahmed. 2016. Model-based gait and action recognition using kinect. *Diss. University of Calgary* (2016).

[3] Fraser Anderson, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2013. YouMove: enhancing movement training with an augmented reality mirror. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 311–320. doi:10.1145/2501988.2502045

[4] Kumar Ashutosh, Tushar Nagarajan, Georgios Pavlakos, Kris Kitani, and Kristen Grauman. 2025. ExpertAF: Expert Actionable Feedback from Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13582–13594.

[5] NSCA-National Strength & Conditioning Association. 2021. *Essentials of strength training and conditioning*. Human kinetics.

[6] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies* 4, 3 (2009), 114–123.

[7] Karlin Bark, Emily Hyman, Frank Tan, Elizabeth Cha, Steven A. Jax, Laurel J. Buxbaum, and Katherine J. Kuchenbecker. 2015. Effects of Vibrotactile Feedback on Human Learning of Arm Motions. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 23, 1 (2015), 51–63. doi:10.1109/TNSRE.2014.2327229

[8] Timothy W. Bickmore and Rosalind W. Picard. 2005. Establishing and maintaining long-term human-computer relationships. *ACM Trans. Comput.-Hum. Interact.* 12, 2 (June 2005), 293–327. doi:10.1145/1067860.1067867

[9] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2021. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 1 (2021), 172–186. doi:10.1109/TPAMI.2019.2929257

[10] Edwin Chau, Jiakun Yu, Cagatay Goncu, and Anusha Withana. 2021. Composite Line Designs and Accuracy Measurements for Tactile Line Tracing on Touch Surfaces. *Proc. ACM Hum.-Comput. Interact.* 5, ISS, Article 491 (Nov. 2021), 17 pages. doi:10.1145/3488536

[11] Ling-Hao Chen, Shunlin Lu, Ailing Zeng, Hao Zhang, Benyou Wang, Ruimao Zhang, and Lei Zhang. 2024. MotionLLM: Understanding Human Behaviors from Human Motions and Videos. arXiv:2405.20340 [cs.CV] https://arxiv.org/abs/2405.20340

[12] Liqi Cheng, Hanze Jia, Lingyun Yu, Yihong Wu, Shuainan Ye, Dazhen Deng, Hui Zhang, Xiao Xie, and Yingcai Wu. 2024. VisCourt: In-Situ Guidance for Interactive Tactic Training in Mixed Reality. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–14.

[13] Christopher Clarke, Doga Cavdir, Patrick Chiu, Laurent Denoue, and Don Kimber. 2020. Reactive Video: Adaptive Video Playback Based on User Motion for Supporting Physical Activity. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 196–208. doi:10.1145/3379337.3415591

[14] Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20, 1 (1960), 37–46.

[15] Fernanda De La Torre, Cathy Mengying Fang, Han Huang, Andrzej Banburski-Fahey, Judith Amores Fernandez, and Jaron Lanier. 2024. LLMR: Real-time Prompting of Interactive Worlds using Large Language Models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 600, 22 pages. doi:10.1145/3613904.3642579

[16] Ginger Delmas, Philippe Weinzaepfel, Francesc Moreno-Noguer, and Grégory Rogez. 2023. Posefix: correcting 3d human poses with natural language. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15018–15028.

[17] Eric Demers, Jonathan Pendenza, Valentin Radevich, and Richard Preuss. 2018. The effect of stance width and anthropometrics on joint range of motion in the lower extremities during a back squat. *International journal of exercise science* 11, 1 (2018), 764.

[18] Runlin Duan, Yuzhao Chen, Rahul Jain, Yichen Hu, Jingyu Shi, and Karthik Ramani. 2025. Canvas3D: Empowering Precise Spatial Control for Image Generation with Constraints from a 3D Virtual Canvas. *arXiv preprint arXiv:2508.07135* (2025).

[19] Runlin Duan, Chenfei Zhu, Yuzhao Chen, Yichen Hu, Jingyu Shi, and Karthik Ramani. 2025. DesignFromX: Empowering Consumer-Driven Design Space Exploration through Feature Composition of Referenced Products. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference*. 1040–1060.

[20] Runlin Duan, Chenfei Zhu, Yuzhao Chen, Dizhi Ma, Jingyu Shi, Ziyi Liu, and Karthik Ramani. 2025. SketchConcept: Sketching-based Concept Recomposition for Product Design using Generative AI. *arXiv preprint arXiv:2508.07141* (2025).

[21] Karl Erickson and Jean Côté. 2016. A season-long examination of the intervention tone of coach–athlete interactions and athlete development in youth sport. *Psychology of sport and exercise* 22 (2016), 264–272.

[22] Felix Escalona, Ester Martinez-Martin, Edmanuel Cruz, Miguel Cazorla, and Francisco Gomez-Donoso. 2020. EVA: EVAluating at-home rehabilitation exercises using augmented reality and low-cost sensors. *Virtual Reality* 24, 4 (2020), 567–581. doi:10.1007/s10055-019-00419-4

[23] Yao Feng, Jing Lin, Sai Kumar Dwivedi, Yu Sun, Priyanka Patel, and Michael J Black. 2024. Chatpose: Chatting about 3d human pose. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2093–2103.

[24] Mihai Fieraru, Mihai Zanfir, Silviu Cristian Pirlea, Vlad Olaru, and Cristian Sminchisescu. 2021. AIFit: Automatic 3D Human-Interpretable Feedback Models for Fitness Training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9919–9928.

[25] Norman M. Fraser and G.Nigel Gilbert. 1991. Simulating speech systems. *Computer Speech & Language* 5, 1 (1991), 81–99. doi:10.1016/0885-2308(91)90019-M

[26] Carl Gabbard. 2021. *Lifelong motor development*. Lippincott Williams & Wilkins.

[27] Ann M Gentile. 2000. Skill acquisition: Action, movement, and neuromotor processes. *Movement science* (2000), 111–187.

[28] Google. 2025. NotebookLM. https://notebooklm.google/. Accessed: 2025-09-10.

[29] Google DeepMind Research Team. 2025. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. arXiv:2507.06261 [cs.CL] https://arxiv.org/abs/2507.06261

[30] Yiwen Gu, Shreya Pandit, Elham Saraee, Timothy Nordahl, Terry Ellis, and Margrit Betke. 2019. Home-based physical therapy with an interactive computer vision system. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*. 0–0.

[31] Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human factors* 53, 5 (2011), 517–527.

[32] John Hattie and Helen Timperley. 2007. The power of feedback. *Review of educational research* 77, 1 (2007), 81–112.

[33] Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework. arXiv:2308.00352 [cs.AI] https://arxiv.org/abs/2308.00352

[34] Xiyun Hu, Dizhi Ma, Fengming He, Zhengzhe Zhu, Shao-Kang Hsia, Chenfei Zhu, Ziyi Liu, and Karthik Ramani. 2025. GesPrompt: Leveraging Co-Speech Gestures to Augment LLM-Based Interaction in Virtual Reality. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference (DIS '25)*. Association for Computing Machinery, New York, NY, USA, 59–80. doi:10.1145/3715336.3735769

[35] Keiichi Ihara, Kyzyl Monteiro, Mehrad Faridan, Rubaiat Habib Kazi, and Ryo Suzuki. 2025. Video2MR: Automatically Generating Mixed Reality 3D Instructions by Augmenting Extracted Motion from 2D Videos. In *Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI '25)*. Association for Computing Machinery, New York, NY, USA, 1548–1563. doi:10.1145/3708359.3712159

[36] Atsuki Ikeda, Yuka Tanaka, Dong-Hyun Hwang, Homare Kon, and Hideki Koike. 2019. Golf training system using sonification and virtual shadow. In *ACM SIGGRAPH 2019 Emerging Technologies* (Los Angeles, California) *(SIGGRAPH '19)*. Association for Computing Machinery, New York, NY, USA, Article 14, 2 pages. doi:10.1145/3305367.3327993

[37] Hye-Young Jo, Laurenz Seidel, Michel Pahud, Mike Sinclair, and Andrea Bianchi. 2023. FlowAR: How Different Augmented Reality Visualizations of Online Fitness Videos Support Flow for At-Home Yoga Exercises. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 469, 17 pages. doi:10.1145/3544548.3580897

[38] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. 2018. End-to-End Recovery of Human Shape and Pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[39] J. F. Kelley. 1984. An iterative design methodology for user-friendly natural language office information applications. *ACM Trans. Inf. Syst.* 2, 1 (Jan. 1984), 26–41. doi:10.1145/357417.357420

[40] Dennis Landin. 1994. The role of verbal cues in skill learning. *Quest* 46, 3 (1994), 299–313.

[41] Hsuanling Lee, Jiakun Yu, Shurui Zheng, Te-Yan Wu, and Liang He. 2025. FluxLab: Creating 3D Printable Shape-Changing Devices with Integrated Deformation Sensing. *arXiv preprint arXiv:2512.02911* (2025).

[42] Tinghui Li, Danyang Peng, Eduardo Velloso, Anusha Withana, Kouta Minamizawa, and Zhanna Sarsenbayeva. 2025. Estimating the Effects of Ambient Noise on Mixed Reality Interaction. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 4, Article 189 (Dec. 2025), 34 pages. doi:10.1145/3770715

[43] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. 2023. Univtg: Towards unified video-language temporal grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2794–2804.

[44] Tica Lin, Rishi Singh, Yalong Yang, Carolina Nobre, Johanna Beyer, Maurice A. Smith, and Hanspeter Pfister. 2021. Towards an Understanding of Situated AR Visualization for Basketball Free-Throw Training. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 461, 13 pages. doi:10.1145/3411764.3445649

[45] 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 (2024), 3180–3195. doi:10.1109/TVCG.2022.3230855

[46] Ziyi Liu, Zhengzhe Zhu, Lijun Zhu, Enze Jiang, Xiyun Hu, Kylie A Peppler, and Karthik Ramani. 2024. ClassMeta: Designing Interactive Virtual Classmate to Promote VR Classroom Participation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 659, 17 pages. doi:10.1145/3613904.3642947

[47] Andreas R Luft and Manuel M Buitrago. 2005. Stages of motor skill learning. *Molecular neurobiology* 32, 3 (2005), 205–216.

[48] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172* (2019).

[49] Dizhi Ma, Xiyun Hu, Jingyu Shi, Mayank Patel, Rahul Jain, Ziyi Liu, Zhengzhe Zhu, and Karthik Ramani. 2024. avaTTAR: Table Tennis Stroke Training with Embodied and Detached Visualization in Augmented Reality. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 35, 16 pages. doi:10.1145/3654777.3676400

[50] Ji Ma, Jiale Wu, Haoyu Wang, Yanze Zhang, Xiao Xie, Zheng Zhou, Hui Zhang, Jiachen Wang, and Yingcai Wu. 2025. T3Set: A Multimodal Dataset with Targeted Suggestions for LLM-based Virtual Coach in Table Tennis Training. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2* (Toronto ON, Canada) *(KDD '25)*. Association for Computing Machinery, New York, NY, USA, 5686–5697. doi:10.1145/3711896.3737407

[51] Richard Magill and David I Anderson. 2010. *Motor learning and control*. McGraw-Hill Publishing New York.

[52] Richard A Magill. 1994. The influence of augmented feedback on skill learning depends on characteristics of the skill and the learner. *Quest* 46, 3 (1994), 314–327.

[53] Sara Mandic, Rhys Tracy, and Misha Sra. 2023. ARFit: Pose-based Exercise Feedback with Mobile AR. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction* (Sydney, NSW, Australia) *(SUI '23)*. Association for Computing Machinery, New York, NY, USA, Article 45, 3 pages. doi:10.1145/3607822.3618008

[54] Richard E Mayer. 2002. Multimedia learning. In *Psychology of learning and motivation*. Vol. 41. Elsevier, 85–139.

[55] Brad McKay, Julia Hussien, Mary-Anne Vinh, Alexandre Mir-Orefice, Hugh Brooks, and Diane M Ste-Marie. 2022. Meta-analysis of the reduced relative feedback frequency effect on motor learning and performance. *Psychology of Sport and Exercise* 61 (2022), 102165.

[56] NoteGPT. 2025. NoteGPT. https://notegpt.io/. Accessed: 2025-09-10.

[57] Heather L O'Brien and Elaine G Toms. 2010. The development and evaluation of a survey to measure user engagement. *Journal of the American Society for Information Science and technology* 61, 1 (2010), 50–69.

[58] Praneeth Bimsara Perera, Ravindu Madhushan Pushpakumara, Hiroyuki Kajimoto, Arata Jingu, Jürgen Steimle, and Anusha Withana. 2025. eTactileKit: A Toolkit for Design Exploration and Rapid Prototyping of Electro-Tactile Interfaces. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 131, 17 pages. doi:10.1145/3746059.3747796

[59] Rajiv Ranganathan, Mei-Hua Lee, and Chandramouli Krishnan. 2022. Ten guidelines for designing motor learning studies. *Brazilian journal of motor behavior* 16, 2 (2022), 112.

[60] Judith E Rink. 1993. Teaching physical education for learning. (1993).

[61] Jerzy Sadowski, Andrzej Mastalerz, and Tomasz Niznikowski. 2013. Benefits of bandwidth feedback in learning a complex gymnastic skill. *Journal of human kinetics* 37 (2013), 183.

[62] H. Sakoe and S. Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26, 1 (1978), 43–49. doi:10.1109/TASSP.1978.1163055

[63] Elizabeth A Sanli, Jae T Patterson, Steven R Bray, and Timothy D Lee. 2013. Understanding self-controlled motor learning protocols through the self-determination theory. *Frontiers in psychology* 3 (2013), 611.

[64] Richard A Schmidt and Timothy D Lee. 2019. *Motor learning and performance 6th edition with web study guide-loose-leaf edition: From principles to application.* Human Kinetics Publishers.

[65] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems* 36 (2023), 38154–38180.

[66] David E Sherwood. 1988. Effect of bandwidth knowledge of results on movement consistency. *Perceptual and Motor Skills* 66, 2 (1988), 535–542.

[67] Valerie J Shute. 2008. Focus on formative feedback. *Review of educational research* 78, 1 (2008), 153–189.

[68] Roland Sigrist, Georg Rauter, Robert Riener, and Peter Wolf. 2013. Augmented visual, auditory, haptic, and multimodal feedback in motor learning: a review. *Psychonomic bulletin & review* 20, 1 (2013), 21–53.

[69] Laura St. Germain, Brad McKay, Andrew Poskus, Allison Williams, Olena Leshchyshen, Sherry Feldman, Joshua GA Cashaback, and Michael J Carter. 2023. Exercising choice over feedback schedules during practice is not advantageous for motor learning. *Psychonomic Bulletin & Review* 30, 2 (2023), 621–633.

[70] Yu Sun, Qian Bao, Wu Liu, Tao Mei, and Michael J Black. 2023. Trace: 5d temporal regression of avatars with dynamic cameras in 3d environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8856–8866.

[71] Akifumi Takahashi, Yudai Tanaka, Archit Tamhane, Alan Shen, Shan-Yuan Teng, and Pedro Lopes. 2024. Can a Smartwatch Move Your Fingers? Compact and Practical Electrical Muscle Stimulation in a Smartwatch. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 2, 15 pages. doi:10.1145/3654777.3676373

[72] Kosuke Takahashi, Dan Mikami, Mariko Isogawa, Yoshinori Kusachi, and Naoki Saijo. 2019. VR-based Batter Training System with Motion Sensing and Performance Visualization. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 1353–1354. doi:10.1109/VR.2019.8798005

[73] Caitlin Tenison and John R Anderson. 2016. Modeling the distinct phases of skill acquisition. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 42, 5 (2016), 749.

[74] Atima Tharatipyakul, Kenny T. W. Choo, and Simon T. Perrault. 2020. Pose Estimation for Facilitating Movement Learning from Online Videos. In *Proceedings of the 2020 International Conference on Advanced Visual Interfaces* (Salerno, Italy) *(AVI '20)*. Association for Computing Machinery, New York, NY, USA, Article 64, 5 pages. doi:10.1145/3399715.3399835

[75] Eric Van Breda, Stijn Verwulgen, Wim Saeys, Katja Wuyts, Thomas Peeters, and Steven Truijen. 2017. Vibrotactile feedback as a tool to improve motor learning and sports performance: a systematic review. *BMJ open sport & exercise medicine* 3, 1 (2017).

[76] Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2013. MotionMA: motion modelling and analysis by demonstration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1309–1318. doi:10.1145/2470654.2466171

[77] Jiachen Wang, Ji Ma, Kangping Hu, Zheng Zhou, Hui Zhang, Xiao Xie, and Yingcai Wu. 2022. Tac-trainer: a visual analytics system for IoT-based racket sports training. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2022), 951–961.

[78] Jiachen Wang, Ji Ma, Kangping Hu, Zheng Zhou, Hui Zhang, Xiao Xie, and Yingcai Wu. 2023. Tac-Trainer: A Visual Analytics System for IoT-based Racket Sports Training. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2023), 951–961. doi:10.1109/TVCG.2022.3209352

[79] Jianbo Wang, Kai Qiu, Houwen Peng, Jianlong Fu, and Jianke Zhu. 2019. AI Coach: Deep Human Pose Estimation and Analysis for Personalized Athletic Training Assistance. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 374–382. doi:10.1145/3343031.3350910

[80] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. 2020. Deep High-Resolution Representation Learning for Visual Recognition. arXiv:1908.07919 [cs.CV] https://arxiv.org/abs/1908.07919

[81] Yufu Wang, Ziyun Wang, Lingjie Liu, and Kostas Daniilidis. 2024. TRAM: Global Trajectory and Motion of 3D Humans from in-the-wild Videos. In *European Conference on Computer Vision*. Springer, 467–487.

[82] Jiqing Wen, Lauren Gold, Qianyu Ma, and Robert LiKamWa. 2024. Augmented Coach: Volumetric Motion Annotation and Visualization for Immersive Sports Coaching. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. 137–146. doi:10.1109/VR58804.2024.00037

[83] Jian-Jia Weng, Calvin Ku, Jo Chien Wang, Chih-Jen Cheng, Tica Lin, Yu-An Su, Tsung-Hsun Tsai, You-Yi Lin, Lun-Wei Ku, Hung-Kuo Chu, and Min-Chun Hu. 2025. Bridging Coaching Knowledge and AI Feedback to Enhance Motor Learning in Basketball Shooting Mechanics Through a Knowledge-Based SOP Framework. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 989, 20 pages. doi:10.1145/3706598.3713324

[84] Carolee J Winstein and Richard A Schmidt. 1990. Reduced frequency of knowledge of results enhances motor skill learning. *Journal of Experimental Psychology: Learning, memory, and cognition* 16, 4 (1990), 677.

[85] Craig A Wrisberg and Gabriele Wulf. 1997. Diminishing the effects of reduced frequency of knowledge of results on generalized motor program learning. *Journal of motor behavior* 29, 1 (1997), 17–26.

[86] Yihong Wu, Lingyun Yu, Jie Xu, Dazhen Deng, Jiachen Wang, Xiao Xie, Hui Zhang, and Yingcai Wu. 2023. AR-Enhanced Workouts: Exploring Visual Cues for At-Home Workout Videos in AR Environment. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 121, 15 pages. doi:10.1145/3586183.3606796

[87] Gabriele Wulf and Rebecca Lewthwaite. 2016. Optimizing performance through intrinsic motivation and attention for learning: The OPTIMAL theory of motor learning. *Psychonomic bulletin & review* 23, 5 (2016), 1382–1414.

[88] Gabriele Wulf, Charles H Shea, and Sabine Matschiner. 1998. Frequent feedback enhances complex motor skill learning. *Journal of motor behavior* 30, 2 (1998), 180–192.

[89] Chengshuo Xia, Xinrui Fang, Riku Arakawa, and Yuta Sugiura. 2022. VoLearn: A Cross-Modal Operable Motion-Learning System Combined with Virtual Avatar and Auditory Feedback. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 2, Article 81 (July 2022), 26 pages. doi:10.1145/3534576

[90] Chengshuo Xia, Tian Min, and Yuta Sugiura. 2024. AudioMove: Applying the Spatial Audio to Multi-Directional Limb Exercise Guidance. *Proceedings of the ACM on Human-Computer Interaction* 8, MHCI (2024), 1–26.

[91] Difeng Yu, Mantas Cibulskis, Erik Skjoldan Mortensen, Mark Schram Christensen, and Joanna Bergström. 2024. Metrics of motor learning for analyzing movement mapping in virtual reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–18.

[92] Jiakun Yu, Hasindu Kariyawasam, Shuying Wu, Sriram Subramanian, and Anusha Withana. 2025. Designing Multi-DoF Epidermal Bend Sensors Using Flexible Resistive Traces. *IEEE Sensors Journal* 25, 23 (2025), 42597–42606. doi:10.1109/JSEN.2025.3623068

[93] Jiakun Yu, Supun Kuruppu, Biyon Fernando, Praneeth Bimsara Perera, Yuta Sugiura, Sriram Subramanian, and Anusha Withana. 2024. IrOnTex: Using Ironable 3D Printed Objects to Fabricate and Prototype Customizable Interactive Textiles. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 3, Article 138 (Sept. 2024), 26 pages. doi:10.1145/3678543

[94] Jiakun Yu, Praneeth Bimsara Perera, Rahal Viddusha Perera, Mohammad Mirkhalaf Valashani, and Anusha Withana. 2024. Fabricating Customizable 3-D Printed Pressure Sensors by Tuning Infill Characteristics. *IEEE Sensors Journal* 24, 6 (2024), 7604–7613. doi:10.1109/JSEN.2024.3358330

[95] Running Zhao, Zhihan Jiang, Xinchen Zhang, Chirui Chang, Handi Chen, Weipeng Deng, Luyao Jin, Xiaojuan Qi, Xun Qian, and Edith CH Ngai. 2025. NoteIt: A System Converting Instructional Videos to Interactable Notes Through Multimodal Video Understanding. *arXiv preprint arXiv:2508.14395* (2025).

[96] Hongyu Zhou, Tom Kip, Yihao Dong, Andrea Bianchi, Zhanna Sarsenbayeva, and Anusha Withana. 2025. Juggling Extra Limbs: Identifying Control Strategies for Supernumerary Multi-Arms in Virtual Reality. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 1156, 16 pages. doi:10.1145/3706598.3713647

[97] Qiushi Zhou, Andrew Irlitti, Difeng Yu, Jorge Goncalves, and Eduardo Velloso. 2022. Movement Guidance using a Mixed Reality Mirror. In *Proceedings of the 2022 ACM Designing Interactive Systems Conference* (Virtual Event, Australia) *(DIS '22)*. Association for Computing Machinery, New York, NY, USA, 821–834. doi:10.1145/3532106.3533466

[98] Chenfei Zhu, Shao-Kang Hsia, Xiyun Hu, Ziyi Liu, Jingyu Shi, and Karthik Ramani. 2025. agentAR: Creating Augmented Reality Applications with Tool-Augmented LLM-based Autonomous Agents. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology*. 1–23.

[99] Junyi Zhu, Yuxuan Lei, Aashini Shah, Gila Schein, Hamid Ghaednia, Joseph Schwab, Casper Harteveld, and Stefanie Mueller. 2022. MuscleRehab: Improving Unsupervised Physical Rehabilitation by Monitoring and Visualizing Muscle Engagement. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 33, 14 pages. doi:10.1145/3526113.3545705

A Formative Interview Details

Interview questions:

- Briefly describe your background with this skill and how a typical session is structured.
- Which feedback formats work best for you (e.g., brief verbal cues, demonstration, video replay, metrics)? Why?
- Describe a recent effective feedback episode: what was said/shown, and what changed afterwards?
- What data or visuals (e.g., trajectory, joint angles, speed) make feedback more understandable or actionable?
- When you give/receive feedback, what is the primary intent?
- In what situations do you choose different feedback?
- How do you phrase or interpret feedback so it is immediately actionable on the next repetition?
- When multiple issues occur at once, how do you decide what to address first, and what can wait?
- How do timing and frequency of feedback change as performance improves, plateaus, or regresses?
- Describe a time you changed strategy (e.g., switched cue type, added a task constraint, delayed feedback) and why.

Table 3: Demographics of interviewees, including area, training with coach experience, and coaching experience in years.

ID	Area	Training	Coaching
E1	Yoga	5	1
E2	Tennis	4	0
E3	Soccer	4	6
E4	Table Tennis	22	11
E5	Table Tennis	13	4
E6	Pickleball	4	2
E7	Fitness	5	0
E8	Fitness	10	10

B Video Dataset Statistics for CP-to-Parameter Taxonomy Coding

Table 4: Coding video dataset statistics by domain. #CP denotes the number of coded coaching points (CP) instances extracted from videos

Area	# Video	# CP
Racket Sports	15	58
Fitness	12	47
Bat & Ball Sports	12	47
General Exercises	16	60
Total	55	212

C Reference Videos for User Study

Table 5: Instructional videos used in Phase 1 and Phase 2.

Phase	Skills
Phase 1	Yoga (Trikonasana pose)[1]
	Pickleball (forehand drive)[2]
	Dumbbells (Romanian deadlift)[3]
Phase 2	**Set A**
	Soccer (passing)[4]
	Table tennis (forehand drive)[5]
	Single-leg hip thrust[6]
	Set B
	Breakdance (cross step)[7]
	Tennis (forehand drive)[8]
	Squat[9]

D Word cloud

Figure 14: Word cloud of all chatbox questions.

E Follow-Up Interview Questions for Experienced Learners

- "Thinking about the best human coach you have trained with, what is their feedback typically like? What do you find most helpful about it?"
- "Compared to that coach, how would you describe the quality of AgentCoach's feedback—for example, whether it accurately identifies what you did well or poorly, and gives concrete suggestions to you?"
- "Can you recall a situation where AgentCoach's feedback felt more helpful than what a typical coach would say, and a moment where it felt less helpful?"
- "How does AgentCoach compare to your human coaches in terms of nuance and subtlety—for example, noticing small

[1] https://www.youtube.com/shorts/thybVfw4ZBs
[2] https://www.youtube.com/shorts/SoLgxwo_gM0
[3] https://www.youtube.com/watch?v=hQgFixeXdZo
[4] https://www.youtube.com/watch?v=umYh09GLr7I
[5] https://www.youtube.com/watch?v=5awFA4FGD4k
[6] https://www.youtube.com/shorts/ymg2AMDAwrY
[7] https://www.youtube.com/watch?v=HF-hRLB1cBU
[8] https://www.youtube.com/shorts/gcbh1hKavx0
[9] https://www.youtube.com/watch?v=xqvCmoLULNY

but important details, or phrasing feedback in a calibrated way rather than just 'right' or 'wrong'?"
- When comparing AgentCoach to a human coach, did its feedback feel responsive and adaptive to your performance over time as you practiced? For example, did it reduce corrections when you improved or shift emphasis when an old mistake reappeared?

F Pose Detection Examples

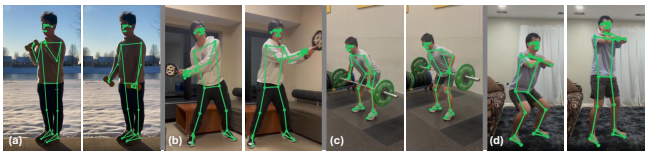

Figure 15: Pose-detection examples across diverse environments (indoor and outdoor) and exercises: (a) dumbbell curls, (b) pickleball forehands, (c) deadlifts, and (d) squats.

G Prompt Template

G.1 Analyzer Prompt

(1) CP-Extraction prompt

```
[1. ROLE & OBJECTIVE]
Act as an expert sports analyst specializing in biomechanics. The mission is to
       analyze exercise videos and generate a structured knowledge base of
       measurable, body-centric coaching cues.

[2. MULTIMODAL INFORMATION HIERARCHY]
Analyze data strictly with the following information:
- AUDIO TRANSCRIPT: Source of truth for phase names, timing, and explicit
       coaching intents.
- VISUAL OVERLAYS: Use on-screen text/graphics if audio is insufficient.
- VISUAL MOVEMENT: Analyze raw body kinematics only when explicit instructions
       are missing.

[3. CUE EXTRACTION PROTOCOL]
- ATOMIC DECOMPOSITION: Complex instructions must be split into single-focus
       cues (e.g., "Bend knees and straight back" --> two separate cues).
- GEOMETRIC PARAMETERIZATION: Map every cue to a quantifiable parameter
       extractable from skeletal data:
    - JointAngle: Angle of a specific joint.
    - LimbPlaneAngle: Angle of a limb segment relative to a plane.
    - Distance: Euclidean distance between two body points.
    - DistanceRatio: Ratio between key anatomic distances (e.g., feet width vs.
       shoulder width).

[4. EXCLUSION CRITERIA (BODY-ONLY CONSTRAINT)]
To ensure computability, strict exclusion rules apply:
- NO EXTERNAL OBJECTS: Discard cues referencing equipment (rackets, balls, nets)
       or floor markers.
- NO SUBJECTIVE SENSATIONS: Discard unquantifiable internal states (e.g., muscle
       activation, pressure) unless translatable to geometry.
- NO ABSTRACT TARGETS: Discard spatial targets not defined by body joints.

[5. OUTPUT SPECIFICATION]
Output shape (JSON object):
{
  "reference_video_id": "{job_id}",
  "exercise_name": "...",
  "overall_summary": "..." | null,
  "overall_transcript": "..." | null,
  "cues": [
    {"name": "...", "description": "...", "detailed_description": "...", "
       reference_start_time": "MM:SS"}
  ]
}

LEARN FROM THESE POSITIVE EXAMPLES:
......
```

(2) Reference-Localization prompt

```
[1. ROLE & OBJECTIVE]
Act as an expert video analyst. The objective is to identify and isolate a
       single, canonical repetition of a specific physical exercise from a raw
       video stream.

[2. CONTEXT]
The system receives an exercise_hint describing the target movement pattern. The
       goal is to find the optimal "Gold Standard" segment to serve as a
       reference for comparison.

[3. SELECTION PROTOCOL (OPTIMALITY CRITERIA)]
Scan the video and select the segment based on the following strict priority
       rules:
- SINGULARITY: The segment must contain exactly one complete repetition cycle.
- VISIBILITY: Prefer segments where the full body is visible throughout the
       entire movement.
- STABILITY: Avoid segments with camera cuts or view changes (e.g., switching
       from front to side view).
- QUALITY: If multiple repetitions exist, select the one that is:
    - Most complete (start to finish).
    - Most unobstructed (no equipment/people blocking the view).
    - Demonstrates the largest Range of Motion (ROM).

[4. OUTPUT SPECIFICATION]
Return a STRICT JSON object (no markdown, no prose) containing the precise
       timestamps and a confidence score.
- TIME FORMAT: Must be strictly MM:SS or HH:MM:SS.
- SCHEMA:
{
  "start_time": "MM:SS",
  "end_time": "MM:SS",
  "confidence": 0.95
}
```

G.2 Builder Prompt

```
[1. ROLE & OBJECTIVE]
Act as a deterministic Biomechanical Parameter Building System. Your objective
       is to translate natural language coaching cues (extracted in the previous
       step) into executable, code-defined biomechanical parameters selected
       strictly from a provided library.

[2. INPUT CONTEXT]
The system receives:
- EXTRACTED CUES: A list of cues containing description and detailed_description.

- PARAMETER LIBRARY: A definitive schema of available calculation modules,
       including available joint vertices, limb segments, and valid measurement
       axes.

[3. MAPPING LOGIC & PROTOCOLS]
- SELECTION CONSTRAINT: You must select 1 to N parameters for each cue. You
       strictly CANNOT invent new parameters; you must choose exclusively from
       the allowed lists provided in the context.
- CONTEXTUAL INFERENCE: Prioritize the detailed_description field to resolve
       ambiguities when selecting the anatomical target (e.g., distinguishing
       between "upper back" and "lower back").
- STABILITY PROTOCOL ("THE MAINTAIN RULE"):
    - If a cue implies consistency or holding a position (e.g., "Keep back
       straight," "Maintain knee width"), you must generate two parameters for
       the same metric: one evaluated at min (minimum peak) and one at max (
       maximum peak). This enforces the value stays within a range throughout
       the movement.

[4. PARAMETER CATEGORY DEFINITIONS]
- JOINTANGLE: Measures the flexion/extension of a specific vertex joint (e.g.,
       left_knee).
- LIMBORIENTATION: Measures the absolute angle of a limb segment relative to a
       reference axis (e.g., torso relative to the vertical axis or anatomical
       lines like shoulder_line).
- DISTANCE: Euclidean distance between any two specified skeletal keypoints (e.g
       ., left_ankle to right_ankle).
- DISTANCERATIO: The ratio between two distances (Presets: feet_vs_shoulders,
       feet_vs_hips).

[5. TEMPORAL EVALUATION POINTS]
Assign exactly one evaluation point to each parameter based on when the cue is
       most critical:
- start / end: For setup or finishing position.
- min / max: For dynamic peaks during the movement (e.g., lowest point of a
       squat).

[6. OUTPUT SPECIFICATION]
OUTPUT SHAPE (JSON ONLY):
{
  "mappings": [
    {
```

```
      "cue_index": 0,
      "parameters": [
        { "type": "joint_angle", "name": "Left Knee Flexion", "vertex_joint": "
       left_knee", "evaluation_point": "min" },
        { "type": "limb_orientation", "name": "Torso vs Screen Y", "limb_segment
       ": ["mid_hip","mid_shoulder"], "evaluation_point": "start" },
        { "type": "distance", "name": "Feet Distance", "joint_pair": ["
       left_ankle","right_ankle"], "evaluation_point": "start" },
        { "type": "distance_ratio", "name": "Feet vs Shoulders Width", "
       joint_pairs": [["left_ankle","right_ankle"],["left_shoulder","
       right_shoulder"]], "evaluation_point": "start" }
      ]
    }
  ]
}

LEARN FROM THESE EXAMPLES:
......
```

G.3 Trainer Prompt

```
[1. ROLE & OBJECTIVE]
Act as an AI Coach based on a state-dependent progression policy. Analyze the
       injected User History (H) and Current Performance Results (R) to produce
       structured, pedagogical feedback.

[2. INPUT CONTEXT SPECIFICATION (RUNTIME INJECTION)]
The system receives a JSON context object containing the following dynamic data.
       (Note: This section is populated at runtime with actual user data)

- HISTORY_BUFFER (H): A chronological list (length <= 5) of previous feedback
       types and verbal outputs.
    - USED FOR: State transition logic and anti-repetition checks.
- CURRENT_METRICS (R): A list of detected errors for the current session,
       including severity levels ("none", "mild", "moderate", "severe") and
       specific biomechanical deviation values.
    - USED FOR: Determining the feedback trigger intensity.

[3. PEDAGOGICAL FEEDBACK TAXONOMY]
Classify feedback into six distinct types (FT):
- TYPE 1 (CORRECTION) / TYPE 2 (EXPLANATION): For errors.
- TYPE 3 (PRAISE): For immediate improvement.
- TYPE 4 (CONFIRMATION) / TYPE 6 (SILENCE): For consistent performance.
- TYPE 5 (REMINDER): For regression.

[4. PROGRESSION POLICY & STATE LOGIC (PP)]
Apply the following decision tree to each cue based on R.severity and H.
       last_feedback_type:

- SCENARIO A: HIGH SEVERITY (NEEDS CORRECTION)
    - If H is empty or previous was Type 3/4/6 --> Type 5 (Reminder) (Regression
       Logic).
    - If previous was Type 1 --> Escalate to Type 2 (Detailed Explanation).
    - Otherwise --> Type 1 (Standard Correction).

- SCENARIO B: LOW SEVERITY (GOOD PERFORMANCE)
    - If previous was Type 1/2 --> Type 3 (Motivational Praise) (Reinforcement
       Logic).
    - If consistent --> Type 4 or Type 6 based on decay schedule.

[5. VERBAL SELECTION STRATEGY]
To manage cognitive load, select max TWO cues for verbalization:
- Prioritize explicit corrections (FT in {1, 2}) and immediate praise (FT=3).
- Use a weighted random selection if multiple cues compete for the same priority
       slot.

[6. LINGUISTIC & STYLISTIC PROTOCOLS]
- TTS OPTIMIZATION: Map technical terms to layperson vocabulary (e.g., "Knee
       Flexion" --> "Knee Bend").
- ANTI-REPETITION: Check H.verbal_history; strictly prohibit repeating sentences
       used in the last 5 turns.

[7. OUTPUT SPECIFICATION]
OUTPUT SCHEMA (JSON ONLY)
{
  "cue_feedback": [
    { "cue_name": string, "feedback": string, "feedback_type": 1|2|3|4|5|6 }
  ],
  "overall_summary": string,
  "verbal_feedback":
```