CLAP: CLOSED-LOOP AGENTIC PERCEPTION UNLOCKS LATENT ECG DIAGNOSTIC CAPABILITY IN MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

The electrocardiogram (ECG) is the most accessible tools for diagnosing cardiovascular disease. Recent advances in multimodal large language models (MLLMs) offer a promising path toward automated ECG diagnosis, yet a gap remains between current performance and clinical reliability. What fundamentally limits their diagnostic capability? In this paper, we reveal a perception–reasoning gap: both ECG-specific and general-purpose MLLMs struggle to reliably ground diagnoses in ECG evidence, while exhibiting stronger diagnostic ability when the relevant evidence is reliably perceived. To unlock this capability, we introduce Closed-Loop Agentic Perception (CLAP), an inference-time framework that turns ECG perception from a passive prediction into an active, verifiable, and iterative process. CLAP combines Perceptual Backprojection, which re-instantiates intermediate perceptual hypotheses on the source ECG for verification and refinement, with ECG Perceptual Scaffolding, which organizes interdependent perceptual tasks and coordinates grounded evidence across them. Experimental results show that, without additional training, CLAP consistently improves diagnostic performance across three general-purpose MLLMs, increasing patient-wise F1 by at least $1.44\times$ and enabling them to approach or surpass SOTA ECG-specific MLLMs. These gains, accompanied by consistent improvements in perceptual fidelity, highlight the broader potential of agentic intelligence to close the perception–reasoning gap toward clinically reliable ECG intelligence.

1 INTRODUCTION

Cardiovascular disease remains the leading cause of death worldwide. The electrocardiogram (ECG) is the most accessible and widely used tools for cardiac diagnosis, making its automated interpretation a longstanding goal of clinical artificial intelligence. Over the past decade, deep learning has substantially advanced this goal, with discriminative models achieving high performance across a wide range of rhythm and morphological abnormalities (Hannun et al., 2019; Ribeiro et al., 2020). More recently, rapid advances in multimodal large language models (MLLMs) (Liu et al., 2023; Zhu et al., 2024a) have opened a new possibility: by connecting visual reading with language and medical knowledge, MLLMs promise to move ECG diagnosis beyond predefined label prediction toward clinically grounded interpretation and reasoning. This promise is being pursued from two directions: developing ECG-specific MLLMs through domain training (Wan et al., 2025; Liu et al., 2026; Jin et al., 2026) and multimodal alignment (Vaid et al., 2023; McKeen et al., 2025; Lan et al., 2025), and applying increasingly capable general-purpose MLLMs directly to ECG interpretation (Günay et al., 2024; Engelstein et al., 2025). Although these advances have significantly expanded the capabilities of automated ECG interpretation, considerable room for improvement remains toward the central clinical goal of reliable ECG diagnosis (Oh et al., 2026).

In this work, we reveal a fundamental perception–reasoning gap in MLLM-based ECG diagnosis. Conceptually, ECG diagnosis requires two distinct capabilities: perceiving clinically relevant evidence from the raw ECG and reasoning over that evidence to reach a diagnosis. Our experiments show that diagnostic performance broadly tracks perceptual fidelity across MLLMs, and that the same models perform substantially better when provided with reliable ECG evidence. This suggests that many diagnostic failures originate upstream in perception, where missed or hallucinated evi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

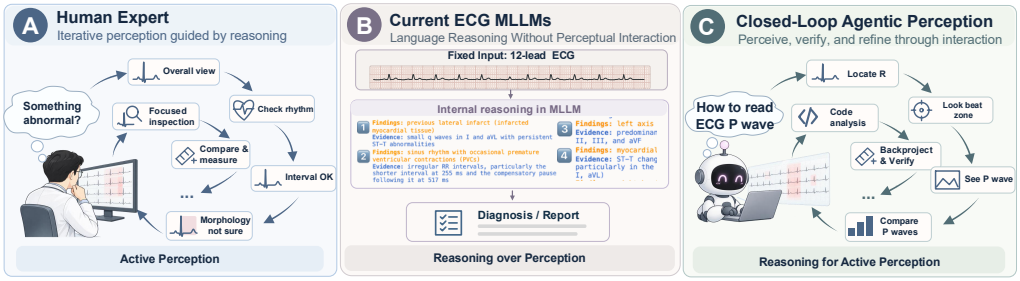


Figure 1: From passive to active ECG perception. (A) Human experts iteratively inspect and refine ECG evidence under the guidance of clinical reasoning. (B) Current ECG MLLMs primarily reason over passively acquired perceptual representations, without explicitly interacting with the ECG to verify and refine intermediate perception. (C) Agentic interaction enables perception itself to enter the loop, allowing perceptual evidence to be actively acquired, verified, and refined against the underlying ECG before diagnosis.

dence can propagate through subsequent reasoning to the final diagnosis. The resulting gap between what MLLMs can diagnose given reliable evidence and what they can diagnose directly from raw ECGs reveals the latent diagnostic capability behind the model. The bottleneck, therefore, is not merely whether MLLMs know how to diagnose ECGs, but whether they can reliably perceive the evidence needed to exercise that knowledge.

Actually, perception is not inherently a passive act of receiving sensory input, but an active process shaped by what we seek to understand (Gottlieb & Oudeyer, 2018; Najemnik & Geisler, 2005). Human experts rarely interpret complex evidence in a single pass. As shown in Fig. 1, when reading an ECG, for example, a cardiologist may notice an irregular rhythm, inspect P waves across beats and leads, and revisit ambiguous waveforms as new evidence emerges. Perception and reasoning thus evolve together through repeated interaction with the underlying signal. Recent advances in AI agents have enabled models to reason, act, and interact with their inputs and environments (Yao et al., 2023; Shinn et al., 2023; Wang et al., 2024). This motivates a different use of agentic intelligence for ECG interpretation, where interaction serves not only to extend reasoning, but to actively shape the perceptual process itself.

However, turning this principle into an effective agentic perception framework raises two central challenges that are particularly pronounced for dynamic biosignals such as ECGs. First, how can an agent verify its ECG perception? Agentic processes depend on evaluative feedback to assess the outcomes of their actions and guide subsequent decisions. Such feedback is readily available in many domains. For example, code can be validated through execution and retrieved information against external sources, whereas ECG perception lacks such an explicit verifier. A measurement may numerically fall within a clinically plausible range while still being poorly grounded in the actual waveform, providing no reliable guidance for subsequent refinement. Second, how can ECG perception be organized into an agentic process? ECG perception is inherently complex, with each perceptual judgment both relying on and constraining others. For example, identifying a P wave may depend jointly on its timing, consistency across beats, and morphology across leads, which may in turn revise the previous perception. Without effective coordination, locally plausible findings can become mutually incompatible, while errors in intermediate perception may be carried forward and amplified through subsequent agentic interactions.

To address these challenges, we introduce Closed-Loop Agentic Perception (CLAP), an inference-time agentic framework that transforms ECG perception from a passive prediction into an active and iterative process. CLAP is built on two key designs. First, we introduce Perceptual Backprojection to close the ECG verification loop. Within each perceptual task, the agent reasons and acts on the ECG to form an intermediate perceptual hypothesis. Rather than treating this hypothesis as an endpoint, Perceptual Backprojection maps it back to its sensory origin, translating a numerical measurement or description into a visual hypothesis instantiated on the original ECG and turning an otherwise unverifiable output into grounded perceptual feedback that directs subsequent refinement. Second, we introduce ECG Perceptual Scaffolding to organize complex and interdependent ECG perception. Based on clinical knowledge, we decomposes ECG interpretation into a sequence of

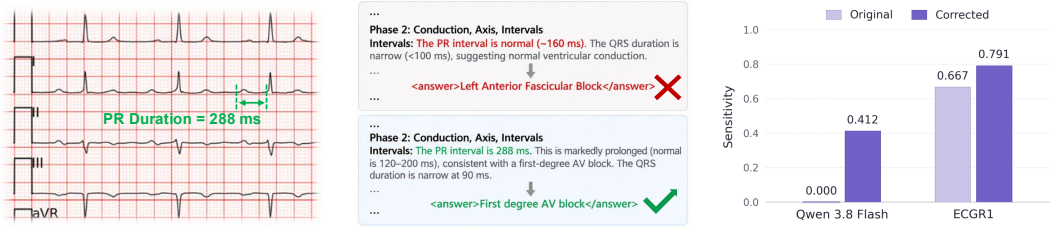
meta-perceptual tasks, while leaving the agent free to determine how each task should be resolved. Within this scaffolding, we introduce a multimodal perceptual ledger that represents each committed task state jointly by its structured output and visually verified representation, allowing grounded perceptual states rather than raw exploration histories to be shared across tasks. Once perception is complete, CLAP provides the accumulated perceptual state for diagnostic reasoning, decoupling evidence perception from diagnosis and enabling general-purpose MLLMs to approach or surpass SOTA ECG-specific MLLMs' performance.

Our work makes the following contributions:

- We reveal a fundamental perception–reasoning gap in MLLM-based ECG diagnosis, showing that unreliable perception masks substantial latent diagnostic capability and identifying perceptual failure as a key bottleneck across MLLMs.
- We introduce Closed-Loop Agentic Perception (CLAP), the first agentic framework to explicitly target the perceptual bottleneck in ECG diagnosis. CLAP couples Perceptual Backprojection, which turns intermediate perceptual hypotheses into grounded feedback for iterative refinement, with ECG Perceptual Scaffolding, which coordinates perception through clinically grounded meta-perceptual tasks and multimodal perceptual memory. Together, these designs substantially improve perceptual fidelity and unlock the downstream diagnostic capability of MLLMs.
- We demonstrate that our agentic perception framework substantially unlocks the diagnostic capability of general-purpose MLLMs. Without additional training, CLAP markedly improves their diagnostic performance, enabling them to approach or surpass SOTA ECG-specific models.

2 Uncovering the Perception–Reasoning Gap

To isolate whether diagnostic errors originate from perception or reasoning, we first perform a controlled intervention on the perceptual evidence while keeping the downstream model unchanged. Figure 2(a–b) shows a representative failure case from Qwen 3.8 Flash. The ECG exhibits first-degree atrioventricular (AV) block, with a markedly prolonged PR interval of 288 ms. When directly interpreting the ECG, the model incorrectly perceives the PR interval as approximately 160 ms and subsequently produces an incorrect diagnosis of left anterior fascicular block. We intervene only at this perceptual step by replacing the erroneous PR measurement with the correct value of 288 ms, truncate the original trajectory at this point, and resume inference from the corrected perceptual state. Without changing the model or providing the diagnostic label, the subsequent reasoning correctly identifies first-degree AV block. This example illustrates that the model can possess the knowledge required for the correct diagnosis while failing to exercise it because the relevant ECG evidence was incorrectly perceived. We next quantify this effect across all first-degree AV block cases by system-



(a) ECG Input (b) Correct Perception Unlocks Diagnosis (c) Perception–Reasoning Gap

Figure 2: The perception–reasoning gap in ECG diagnosis. (a) An ECG with a prolonged PR interval of 288 ms. (b) Correcting the perceived PR interval changes an incorrect diagnosis to first-degree AV block. (c) Sensitivity for first-degree AV block before and after perceptual correction.

atically correcting the key PR-interval evidence while leaving the remaining input and diagnostic reasoning unchanged. As shown in Figure 2(c), Qwen 3.8 Flash has zero sensitivity under direct ECG interpretation, but reaches 0.412 after perceptual correction. ECG-R1, despite being specifically trained for ECG interpretation and already achieving substantially higher sensitivity, further

improves from 0.667 to 0.791 under the same intervention. These results reveal a consistent gap between what MLLMs can diagnose from their own perception and what they can diagnose once the relevant evidence is reliably perceived. We term this discrepancy the *perception–reasoning gap*. Crucially, this gap masks the diagnostic capability of the underlying model: errors introduced during perception propagate into downstream reasoning, making end-to-end diagnostic performance conflate what a model can diagnose with what it can reliably perceive.

3 Closed-Loop Agentic Perception

3.1 Overview

The perception–reasoning gap identified in Section 2 suggests that improving ECG diagnosis does not necessarily require changing what an MLLM knows, but rather how reliably it acquires the evidence on which that knowledge operates. This motivates Closed-Loop Agentic Perception, which restructures ECG perception at two complementary levels, as shown in Fig. 3. At the global level, an ECG perceptual scaffold organizes the overall perception process into clinically grounded meta-perceptual tasks and coordinates evidence across them. At the local level, each task is resolved through an agentic loop that actively interacts with the ECG and iteratively refines perception. Together, these two levels progressively construct the perceptual evidence used for subsequent diagnostic reasoning.

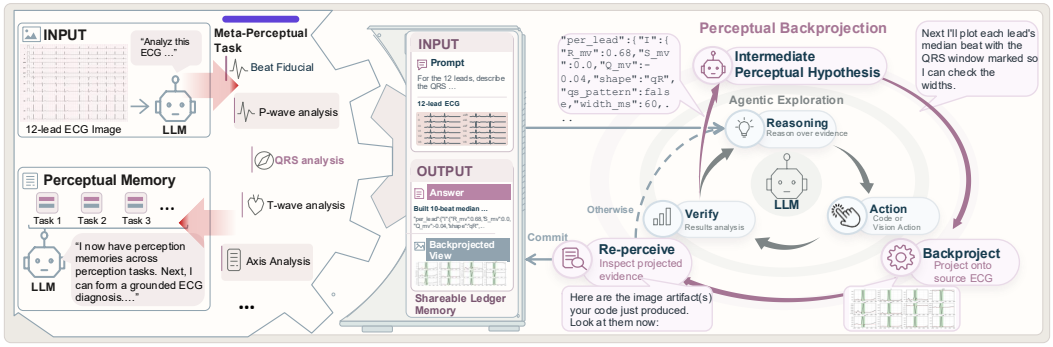


Figure 3: Overview of Closed-Loop Agentic Perception (CLAP). CLAP restructures ECG perception at two complementary levels. At the global level, ECG Perceptual Scaffolding organizes ECG interpretation into clinically grounded meta-perceptual tasks and maintains committed evidence across tasks through a multimodal perceptual ledger. At the local level, each task is resolved through an agentic loop, where Perceptual Backprojection re-instantiates intermediate perceptual hypotheses on the source ECG for visual verification and iterative refinement. The resulting perceptual states are progressively accumulated and finally provided for diagnostic reasoning.

3.2 Closed-Loop Perception via Backprojection

Consider a task T in which an agent interacts with an environment to reach a desired outcome. At step t, the agent receives an observation $o_t \in \mathcal{O}$ and selects an action $a_t \in \mathcal{A}$ according to its current context,

$$a_t \sim \pi(a_t \mid c_t), \qquad c_t = (T, o_1, a_1, \ldots, o_t). \tag{1}$$

Each action produces a new observation that is incorporated into the context and influences subsequent reasoning and action. For this iterative process to refine a solution, the agent must also be able to evaluate whether its current result should be retained or revised. In many agentic tasks, such verification arises naturally from the interaction itself. Code execution exposes errors and test outcomes, while retrieved evidence can support or contradict an intermediate answer.

Such verification is less straightforward for ECG perception. An agent may visually inspect an ECG or analyze it programmatically and obtain a seemingly reasonable measurement or morphological

description. However, a plausible value does not indicate whether the correct waveform boundaries were identified, just as successful code execution does not guarantee that the intended cardiac structure was measured. The agent may therefore know that its result is plausible without knowing whether it is perceptually grounded in the ECG. We refer to this provisional result as an *intermediate perceptual hypothesis* $h \in \mathcal{H}$.

The key idea of CLAP is simple: *perception should be verified in the space where the evidence resides*. A perceptual hypothesis may be expressed numerically or semantically, but its correctness is ultimately determined by its correspondence with the underlying sensory signal. We therefore introduce **Perceptual Backprojection**, which maps an intermediate hypothesis back to its sensory origin,

$$v = \mathcal{B}(X, h), \qquad \mathcal{B} : \mathcal{O} \times \mathcal{H} \to \mathcal{O}, \tag{2}$$

where v is a visual instantiation of the hypothesis on the original ECG. By re-instantiating h on the source ECG, Backprojection $\mathcal{B}$ makes the hypothesis and its supporting evidence jointly perceptible, turning an implicit hypothesis–signal correspondence into an explicit object of verification.

Importantly, Backprojection specifies *what* must become observable rather than *how* it must be rendered. CLAP does not rely on a predefined renderer or an additional trained verifier. Conditioned on T, X, and h, the model determines how the hypothesis should be visually instantiated and programs the corresponding representation, such as waveform reconstruction, fiducial markers, interval boundaries, or lead-specific overlays.

In our implementation, each meta-perceptual task proceeds in the following loops. The agent first explores the task through reasoning, visual inspection, and programmable signal analysis until it forms an intermediate hypothesis $h^{(k)}$. It then backprojects and re-perceives this hypothesis. If the visualized hypothesis is sufficiently supported by the underlying waveform, its structured result and visual grounding are committed as the perceptual output. Otherwise, the observed discrepancy is returned to the agent context, triggering renewed exploration before a revised hypothesis $h^{(k+1)}$ is formed and backprojected again:

$$\textit{explore} \to \textit{hypothesize} \to \textit{backproject} \to \textit{re-perceive} \to \begin{cases} \textit{commit}, & \text{if supported,} \\ \textit{re-explore}, & \text{otherwise.} \end{cases} \tag{3}$$

Perceptual Backprojection therefore does not merely visualize an intermediate result. It turns that result into a new observation that can regulate the next round of perception, closing the loop between perceptual inference and its sensory evidence. Appendix A.5 illustrates this process in a worked case study.

3.3 ECG Perceptual Scaffolding

Closing the loop within an individual perceptual task does not by itself yield a complete ECG interpretation. ECG perception involves multiple interdependent judgments, where evidence established in one perceptual problem may inform, constrain, or trigger re-examination of others. Directly exposing an agent to the entire ECG interpretation problem therefore leaves it to simultaneously decide *what* evidence should be established and *how* each piece of evidence should be obtained. We instead separate these two roles through **ECG Perceptual Scaffolding**.

Specifically, clinical knowledge organizes ECG perception into a sequence of meta-perceptual tasks

$$\mathcal{T} = \{T_1, T_2, \ldots, T_K\}, \tag{4}$$

where each T_i specifies a clinically meaningful perceptual objective and its expected output structure, but leaves the solution procedure unspecified. The scaffold therefore determines *what* should be perceived, while the agent autonomously determines *how* to perceive it through the closed-loop process in Section 3.2. This preserves flexible agentic exploration within each task while imposing clinical structure on the overall perceptual process.

A second challenge is how perceptual evidence should be carried across tasks. Passing the full interaction history would accumulate task-specific reasoning, failed attempts, and uncertain intermediate hypotheses, allowing local exploration to contaminate subsequent perception. CLAP instead maintains a multimodal perceptual ledger memory that records only committed perceptual evidence.

After resolving T_i, the agent commits a structured finding s_i together with its visual grounding v_i, and updates

$$M_i = M_{i-1} \cup \{(s_i, v_i)\}, \tag{5}$$

where M_i denotes the perceptual state available to subsequent tasks. The ledger therefore communicates not the trajectory used to reach a perception, but *what was perceived and where that perception is grounded*. Each task can consequently begin with a compact, multimodally grounded perceptual state while retaining an independent context for its own exploration.

This scaffold turns a collection of local perceptual loops into a coherent ECG interpretation process. Subsequent tasks can build on previously committed evidence, while uncertain exploration remains local to the task in which it occurs. After all meta-perceptual tasks are completed, the accumulated perceptual state M_K is integrated and provided to the MLLM together with the original ECG for diagnostic reasoning,

$$\hat{y} = \mathcal{D}(X, M_K), \tag{6}$$

where $\mathcal{D}$ denotes the diagnostic reasoning process. Importantly, diagnostic labels are not introduced during perceptual construction. CLAP therefore separates the construction of ECG evidence from reasoning over its diagnostic implications, allowing improvements in perception to be evaluated independently from downstream diagnosis. We provide a case study illustrating this workflow in Appendix A.5.

4 Evaluation Protocol

4.1 Evaluation Tasks and Metrics

We center our evaluation on the two capabilities that directly determine the clinical utility of ECG interpretation: whether a model can faithfully perceive the physiological evidence encoded in an ECG, and whether it can translate that evidence into the correct diagnosis. This formulation deliberately evaluates the clinically relevant endpoints of ECG interpretation rather than properties of the generated language itself.

Diagnostic performance. MLLMs generate open-ended diagnostic descriptions whose terminology can vary substantially across models. We therefore normalize both model predictions and reference diagnoses according to the AHA Recommendations for the Standardization and Interpretation of the Electrocardiogram (Mason et al., 2007), mapping free-form outputs into 106 clinically defined primary diagnostic conditions. Performance is evaluated at both the patient and dataset levels. At the patient level, we report patient-wise F1. At the dataset level, we report F1, precision, and recall to characterize overall diagnostic performance. We use F1 as the primary metric because it is well suited to the imbalanced distribution of ECG conditions. Detailed label definitions, normalization procedures, and metric formulations are provided in Appendix A.2.

Perceptual fidelity. We evaluate ECG perception through clinically interpretable measurements that can be quantitatively referenced to the underlying signal. Specifically, we assess seven core perceptual attributes: heart rate (HR), P-wave duration, PR interval, QRS duration, QT interval, corrected QT interval (QTc), and mean QRS axis. Together, these measurements capture fundamental rhythm, timing, conduction, and electrical-axis information that underlies a broad range of ECG diagnoses. Because these measurements differ in scale and units, we evaluate each attribute using the Pearson correlation between model predictions and reference measurements. We then aggregate the correlations across attributes using a mean average to obtain the overall Perceptual Fidelity score. The groundtruth construction and complete metric definitions are detailed in Appendix A.3.

4.2 Datasets, Models, and Baselines

The evaluation is conducted on test sets from two datasets. Following ECG-R1 (Jin et al., 2026) and GEM (Lan et al., 2025), we use 2,381 samples from MIMIC-IV-ECG (Gow et al., 2023; Pollard et al., 2026) as our primary evaluation set. Since its natural diagnostic distribution is highly imbalanced, we further construct a condition-balanced evaluation set from SPH (Liu et al., 2022) to provide broader coverage across diagnostic conditions. Specifically, we randomly sample up to

30 ECGs per condition from SPH, retaining all available cases when fewer than 30 are present, resulting in 531 ECGs in total. Detailed dataset statistics and diagnostic distributions are provided in Appendix A.4.

As CLAP is a model-agnostic agentic framework, we evaluate it with three general-purpose MLLMs as backbones: DeepSeek 4.1 Flash (DeepSeek, 2026), Qwen 3.8 Flash (Alibaba Cloud, 2026), and Claude Opus 5 (Anthropic, 2026). For comparison, we include three ECG-specific MLLMs, ECG-R1 (Jin et al., 2026), GEM (Lan et al., 2025), and PULSE (Liu et al., 2026), as well as representative agentic methods, including ReAct (Yao et al., 2023), CodeAct (Wang et al., 2024), Reflexion (Shinn et al., 2023), and ActiView (Wang et al., 2025). CLAP and all agentic baselines are instantiated on general-purpose MLLMs, with built-in reasoning disabled or set to the minimum level supported by each model. All methods are evaluated without additional training under the same inference setting.

5 Results

5.1 CLAP Unlocks Diagnostic Capability

Tables 1 and 2 report diagnostic performance on MIMIC-TEST and SPH-TEST. On MIMIC-TEST, ECG-specific MLLMs remain strong, reflecting the benefit of domain-specific training on an in-distribution benchmark. Nevertheless, CLAP substantially closes this gap without additional training. With Claude Opus 5 as the backbone, CLAP achieves a patient-wise F1 of 0.710, essentially approaching the best ECG-specific model (ECG-R1, 0.740), while Qwen 3.8 Flash reaches 0.641. More importantly, CLAP consistently improves each general-purpose backbone over direct inference: patient-wise F1 increases from 0.251 to 0.468 for DeepSeek 4.1 Flash, from 0.302 to 0.641 for Qwen 3.8 Flash, and from 0.262 to 0.710 for Claude Opus 5, corresponding to $1.86\times$, $2.12\times$, and $2.71\times$ improvements, respectively.

Table 1: Diagnostic performance on MIMIC.

Model	Patient-wise	Micro-averaged			Macro-averaged		
	F1	F1	Precision	Recall	F1	Precision	Recall
General-Purpose MLLMs							
Claude Opus 5	0.262	0.243	0.242	0.244	0.064	0.098	0.071
Qwen 3.8 Flash	0.302	0.250	0.260	0.241	0.100	0.133	0.080
DeepSeek 4.1 Flash	0.251	0.193	0.280	0.147	0.029	0.048	0.021
ECG-Specific MLLMs							
ECG-R1	**0.740**	**0.691**	**0.751**	0.639	0.243	0.286	0.241
GEM	0.672	0.601	0.527	**0.699**	0.293	0.343	0.308
PULSE	0.711	0.680	0.666	0.694	**0.337**	**0.385**	**0.343**
CLAP							
DeepSeek 4.1 Flash	0.468	0.425	0.445	0.407	0.153	0.257	0.155
Qwen 3.8 Flash	0.641	0.567	0.592	0.544	0.190	0.242	0.213
Claude Opus 5	0.710	0.660	0.680	0.641	0.305	0.350	0.270

Table 2: Diagnostic performance on SPH.

Model	Patient-wise	Micro-averaged			Macro-averaged		
	F1	F1	Precision	Recall	F1	Precision	Recall
General-Purpose MLLMs							
Claude Opus 5	0.267	0.302	0.262	0.355	0.290	0.388	0.321
Qwen 3.8 Flash	0.107	0.120	0.111	0.130	0.079	0.138	0.102
DeepSeek 4.1 Flash	0.009	0.014	0.021	0.011	0.014	0.068	0.009
ECG-Specific MLLMs							
ECG-R1	0.266	0.331	0.344	0.319	0.223	0.252	0.254
GEM	0.256	0.302	0.237	0.414	0.248	0.278	0.330
PULSE	0.284	0.310	0.271	0.362	0.250	0.289	0.296
CLAP							
DeepSeek 4.1 Flash	0.257	0.293	0.263	0.329	0.234	0.343	0.289
Qwen 3.8 Flash	0.349	0.361	0.324	0.409	0.293	0.399	0.343
Claude Opus 5	**0.385**	**0.429**	**0.375**	**0.501**	**0.420**	**0.523**	**0.453**

The advantage becomes more pronounced on SPH-TEST, where none of the compared ECG-specific models is trained on the target dataset. Their patient-wise F1 drops to 0.256–0.284, indicating substantial degradation under the shifted data distribution. In contrast, CLAP with Claude Opus 5 achieves the best overall performance at 0.385, while CLAP with Qwen 3.8 Flash reaches 0.349 and also surpasses all ECG-specific models. Even DeepSeek 4.1 Flash, whose direct performance is only 0.009, reaches 0.257 with CLAP, comparable to ECG-specific models. Relative to direct inference, CLAP improves patient-wise F1 by $28.6\times$, $3.26\times$, and $1.44\times$ for DeepSeek, Qwen, and Opus, respectively. These results show that CLAP does not merely improve an already capable model: it can recover substantial diagnostic capability that remains inaccessible during direct ECG interpretation, while retaining strong generalization across datasets.

5.2 Diagnostic Gains Track Perceptual Fidelity

We next examine whether the diagnostic gains of CLAP are associated with improved ECG perception. Figure 4 jointly compares patient-wise F1 and Perceptual Fidelity on SPH-TEST. Across general-purpose MLLMs, direct diagnostic performance closely follows perceptual fidelity: DeepSeek exhibits both weak perception (0.225) and near-zero diagnostic F1 (0.009), whereas Opus shows substantially stronger perception (0.575) together with higher diagnostic performance (0.267). This relationship persists after applying CLAP. Perceptual Fidelity rises to 0.541, 0.539, and 0.756 for DeepSeek, Qwen, and Opus, respectively, accompanied by consistent gains in diagnosis.

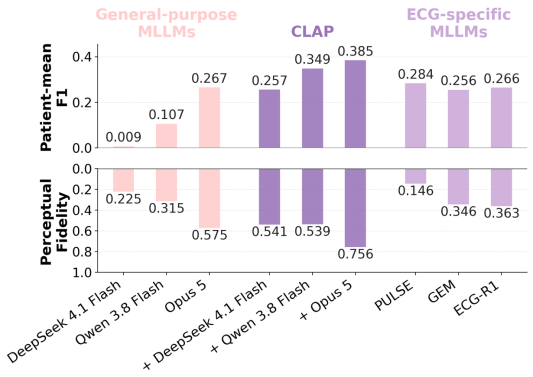


Figure 4: Diagnostic performance and perceptual fidelity on SPH-TEST. Patient-wise F1 (top) and Perceptual Fidelity (bottom) are compared across general-purpose MLLMs, their CLAP counterparts, and ECG-specific MLLMs.

Notably, CLAP raises the perceptual fidelity of all three general-purpose MLLMs to levels comparable to or above the ECG-specific models. CLAP with Opus reaches the highest Perceptual Fidelity (0.756) and simultaneously the highest diagnostic performance (0.385). The aligned improvement in perception and diagnosis across backbones provides empirical support for the perception–reasoning gap identified in Section 2: improving how reliably an MLLM perceives ECG evidence can expose diagnostic capability that is otherwise masked during direct interpretation.

5.3 Generalization Across Diagnostic Conditions

Figure 5 further breaks down performance by diagnostic condition on SPH-TEST. Direct general-purpose MLLMs exhibit highly uneven coverage: DeepSeek and Qwen fail to recognize many conditions, while Opus provides broader but still heterogeneous performance. CLAP substantially expands this diagnostic coverage across all three backbones. In particular, CLAP with Opus achieves the strongest performance across a broad range of rhythm, conduction, axis, chamber, and ST–T abnormalities, while Qwen and DeepSeek recover diagnostic capability for many conditions on which their direct predictions are weak or entirely unsuccessful.

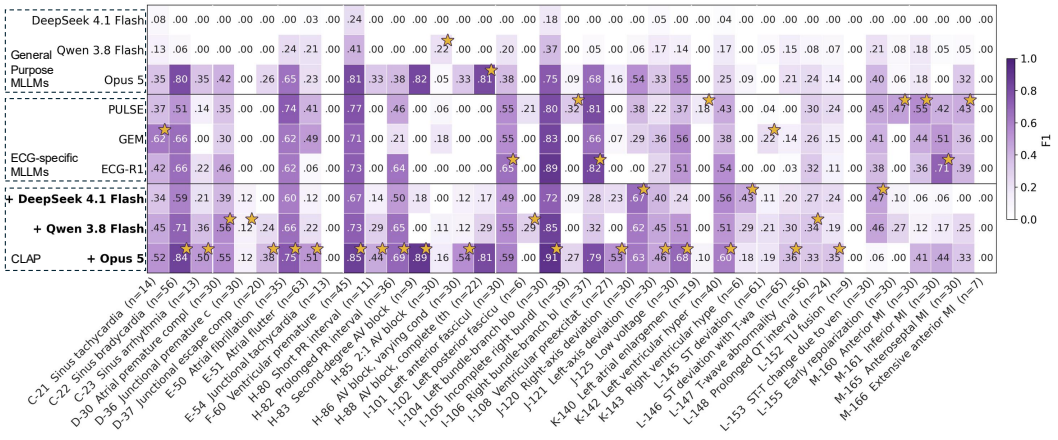


Figure 5: Condition-level diagnostic F1 on SPH-TEST across general-purpose MLLMs, ECG-specific MLLMs, and CLAP. Columns correspond to diagnostic conditions and rows to models; darker shading indicates higher F1.

The condition-level results also reveal complementary strengths and remaining limitations. ECG-specific models retain advantages for several highly morphology-dependent conditions, most notably myocardial infarction subtypes, where domain-specific training remains beneficial. In contrast, their performance varies considerably across other conditions under SPH-TEST, whereas CLAP exhibits broader cross-condition coverage without dataset-specific training. These results suggest that closed-loop perception provides a more general route to transferring the latent clinical knowledge of general-purpose MLLMs across heterogeneous ECG abnormalities, while fine-grained morphology, particularly infarction patterns, remains an important direction for further improving agentic ECG perception.

5.4 Agentic Comparison and Ablation

We first examine whether the gains of CLAP can be attributed simply to generic agentic interaction. Using the same Qwen 3.8 Flash backbone, CLAP substantially outperforms representative agentic methods in both perception and diagnosis (Figure 6). The consistent advantage in both dimensions suggests that additional reasoning, reflection, or tool interaction alone is insufficient; explicitly directing agentic interaction toward perceptual verification and refinement is critical for improving ECG diagnosis. We further ablate the key components of CLAP to identify where these gains arise (Table 3). Removing the perceptual ledger reduces patient-wise F1 from 0.349 to 0.310, while removing Perceptual Backprojection causes a larger drop to 0.270. When both components are removed, leaving the underlying agentic scaffold alone, performance further decreases to 0.210, demonstrating the importance of both local perceptual verification and cross-task evidence coordination.



Figure 6: CLAP versus agentic baselines.

Table 3: Component ablations.

	Full CLAP	Without ledger summary	Without back-projection	Framework only
Patient-wise F1	**0.349**	0.310	0.270	0.210
Micro-averaged F1	**0.361**	0.321	0.279	0.217
Micro-averaged Precision	**0.324**	0.288	0.251	0.194
Micro-averaged Recall	**0.409**	0.363	0.316	0.245
Macro-averaged F1	**0.293**	0.260	0.227	0.176
Macro-averaged Precision	**0.399**	0.354	0.309	0.239
Macro-averaged Recall	**0.343**	0.305	0.265	0.206

6 Conclusion

We uncover a perception–reasoning gap in ECG diagnosis, showing that unreliable perception can mask diagnostic capability already present in MLLMs. Building on this insight, we introduce CLAP, which makes perception active, verifiable, and iterative, substantially improving both perceptual fidelity and diagnosis without additional training. More broadly, our results suggest that MLLM capability should not be bounded by its initial perception. By bringing reasoning back to sensory evidence to verify and refine what is perceived, agentic intelligence shifts from reasoning over perception to reasoning for perception, offering a promising direction for reliable multimodal reasoning beyond ECG.

AI Use Statement

Generative AI tools were used for language editing and implementation assistance. AI systems used within our framework, experiments, and baselines constitute part of the research methodology and are documented in the corresponding sections. All research design, analyses, and conclusions were determined and verified by the authors.

REFERENCES

Alibaba Cloud. Alibaba Cloud Model Studio: qwen3.8-flash. Alibaba Cloud Documentation Center, 2026. URL https://www.alibabacloud.com/help/en/model-studio/qwen3-8-flash. Official model documentation. Accessed September 26, 2026.

Anthropic. Introducing Claude Opus 5. Anthropic, July 2026. URL https://www.anthropic.com/news/claude-opus-5. Official release announcement, July 24, 2026. Accessed September 26, 2026.

DeepSeek. DeepSeek-V4.1-Flash: Smarter, faster, more efficient. DeepSeek API Docs, September 2026. URL https://api-docs.deepseek.com/news/news260910/. Official release announcement, September 10, 2026. Accessed September 26, 2026.

Mengjie Deng, Guanting Dong, and Zhicheng Dou. ToolScope: An agentic framework for vision-guided and long-horizon tool use. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 211–225. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.11. URL https://aclanthology.org/2026.findings-acl.11/.

Haya Engelstein, Roni Ramon-Gonen, Avi Sabbag, Eyal Klang, Karin Sudri, Michal Cohen-Shelly, and Israel Barbash. Effectiveness of the GPT-4o model in interpreting electrocardiogram images for cardiac diagnostics: Diagnostic accuracy study. *JMIR AI*, 4:e74426, 2025. doi: 10.2196/74426. URL https://ai.jmir.org/2025/1/e74426/.

Jacqueline Gottlieb and Pierre-Yves Oudeyer. Towards a neuroscience of active sampling and curiosity. *Nature Reviews Neuroscience*, 19(12):758–770, 2018. doi: 10.1038/s41583-018-0078-0. URL https://www.nature.com/articles/s41583-018-0078-0.

Brian Gow, Tom Pollard, Larry A. Nathanson, Alistair Johnson, Benjamin Moody, Chrystinne Fernandes, Nathaniel Greenbaum, Jonathan W. Waks, Parastou Eslami, Tanner Carbonati, Ashish Chaudhari, Elizabeth Herbst, Dana Moukheiber, Seth Berkowitz, Roger Mark, and Steven Horng. MIMIC-IV-ECG: Diagnostic electrocardiogram matched subset. *PhysioNet*, September 2023. doi: 10.13026/4nqg-sb35. URL https://physionet.org/content/mimic-iv-ecg/1.0/. Version 1.0.

Serkan Günay, Ahmet Öztürk, and Yavuz Yiğit. The accuracy of Gemini, GPT-4, and GPT-4o in ECG analysis: A comparison with cardiologists and emergency medicine specialists. *The American Journal of Emergency Medicine*, 84:68–73, 2024. doi: 10.1016/j.ajem.2024.07.043. URL https://doi.org/10.1016/j.ajem.2024.07.043.

Awni Y. Hannun, Pranav Rajpurkar, Masoumeh Haghpanahi, Geoffrey H. Tison, Codie Bourn, Mintu P. Turakhia, and Andrew Y. Ng. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature Medicine*, 25(1):65–69, 2019. doi: 10.1038/s41591-018-0268-3. URL https://www.nature.com/articles/s41591-018-0268-3.

Guillermo Jimenez-Perez, Juan Acosta, Alejandro Alcaine, and Oscar Camara. Generalising electrocardiogram detection and delineation: training convolutional neural networks with synthetic data augmentation. *Frontiers in Cardiovascular Medicine*, 11:1341786, 2024. doi: 10.3389/fcvm.2024.1341786. URL https://www.frontiersin.org/journals/cardiovascular-medicine/articles/10.3389/fcvm.2024.1341786/full.

Jiarui Jin, Haoyu Wang, Xingliang Wu, Xiaocheng Fang, Xiang Lan, Zihan Wang, Deyun Zhang, Bo Liu, Yingying Zhang, Xian Wu, Hongyan Li, and Shenda Hong. ECG-R1: Protocol-guided and modality-agnostic MLLM for reliable ECG interpretation. *arXiv preprint arXiv:2602.04279*, 2026. URL https://arxiv.org/abs/2602.04279.

Xiang Lan, Feng Wu, Kai He, Qinghao Zhao, Shenda Hong, and Mengling Feng. GEM: Empowering MLLM for grounded ECG understanding with time series and images. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/879832148694832be2b4d658744ba72-Abstract-Conference.html.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

Hui Liu, Dan Chen, Da Chen, Xiyu Zhang, Huijie Li, Lipan Bian, Minglei Shu, and Yinglong Wang. A large-scale multi-label 12-lead electrocardiogram database with standardized diagnostic statements. *Scientific Data*, 9(1):272, 2022. doi: 10.1038/s41597-022-01403-5. URL https://www.nature.com/articles/s41597-022-01403-5.

Ruoqi Liu, Yuelin Bai, Xiang Yue, and Ping Zhang. Teaching multimodal LLMs to comprehend 12-lead electrocardiographic images. *npj Digital Medicine*, 9:349, 2026. doi: 10.1038/s41746-026-02551-3. URL https://www.nature.com/articles/s41746-026-02551-3.

Shilong Liu, Hao Cheng, Haotian Liu, Hao Zhang, Feng Li, Tianhe Ren, Xueyan Zou, Jianwei Yang, Hang Su, Jun Zhu, Lei Zhang, Jianfeng Gao, and Chunyuan Li. LLaVA-Plus: Learning to use tools for creating multimodal agents. In *Computer Vision – ECCV 2024*, volume 15105 of *Lecture Notes in Computer Science*, pp. 126–142. Springer Nature Switzerland, 2025. doi: 10.1007/978-3-031-72970-6_8. URL https://doi.org/10.1007/978-3-031-72970-6_8.

Jay W. Mason, E. William Hancock, and Leonard S. Gettes. Recommendations for the standardization and interpretation of the electrocardiogram: Part II: Electrocardiography diagnostic statement list. *Circulation*, 115(10):1325–1332, 2007. doi: 10.1161/CIRCULATIONAHA.106.180201. URL https://doi.org/10.1161/CIRCULATIONAHA.106.180201.

Kaden McKeen, Sameer Masood, Augustin Toma, Barry Rubin, and Bo Wang. ECG-FM: an open electrocardiogram foundation model. *JAMIA Open*, 8(5):ooaf122, 2025. doi: 10.1093/jamiaopen/ooaf122. URL https://doi.org/10.1093/jamiaopen/ooaf122.

Jiri Najemnik and Wilson S. Geisler. Optimal eye movement strategies in visual search. *Nature*, 434:387–391, 2005. doi: 10.1038/nature03390. URL https://www.nature.com/articles/nature03390.

Jungwoo Oh, Hyunseung Chung, Junhee Lee, Min-Gyu Kim, Hangyul Yoon, Ki Seong Lee, Youngchae Lee, Muhan Yeo, and Edward Choi. ECG-Reasoning-Benchmark: A benchmark for evaluating clinical reasoning capabilities in ECG interpretation. *arXiv preprint arXiv:2603.14326*, 2026. URL https://arxiv.org/abs/2603.14326.

Tom Pollard, Benjamin E. Moody, Li-wei H. Lehman, Brian J. Gow, Chrystinne Fernandes, Chen Xie, Alistair Johnson, Roger G. Mark, and Thomas Heldt. PhysioNet as a global platform for biomedical research. *Nature Health*, 1(8):792–795, 2026. doi: 10.1038/s44360-026-00096-z. URL https://www.nature.com/articles/s44360-026-00096-z.

Antônio H. Ribeiro, Manoel Horta Ribeiro, Gabriela M. M. Paixão, Derick M. Oliveira, Paulo R. Gomes, Jéssica A. Canazart, Milton P. S. Ferreira, Carl R. Andersson, Peter W. Macfarlane, Wagner Meira, Jr., Thomas B. Schön, and Antonio Luiz P. Ribeiro. Automatic diagnosis of the 12-lead ECG using a deep neural network. *Nature Communications*, 11(1):1760, 2020. doi: 10.1038/s41467-020-15432-4. URL https://www.nature.com/articles/s41467-020-15432-4.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Akhil Vaid, Joy Jiang, Ashwin Sawant, Stamatios Lerakis, Edgar Argulian, Yuri Ahuja, Joshua Lampert, Alexander Charney, Hayit Greenspan, Jagat Narula, Benjamin Glicksberg, and Girish N. Nadkarni. A foundational vision transformer improves diagnostic performance for electrocardiograms. *npj Digital Medicine*, 6(1):108, 2023. doi: 10.1038/s41746-023-00840-9. URL https://www.nature.com/articles/s41746-023-00840-9.

Zhongwei Wan, Che Liu, Xin Wang, Chaofan Tao, Hui Shen, Jing Xiong, Rossella Arcucci, Huaxiu Yao, and Mi Zhang. MEIT: Multimodal electrocardiogram instruction tuning on large language models for report generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 14510–14527. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.749. URL https://aclanthology.org/2025.findings-acl.749/.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better LLM agents. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 50208–50232. PMLR, 2024. URL https://proceedings.mlr.press/v235/wang24h.html.

Ziyue Wang, Chi Chen, Fuwen Luo, Yurui Dong, Yuanchi Zhang, Yuzhuang Xu, Xiaolong Wang, Peng Li, and Yang Liu. ActiView: Evaluating active perception ability for multimodal large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7605–7633. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.376. URL https://aclanthology.org/2025.acl-long.376/.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wu_V_Guided_Visual_Search_as_a_Core_Mechanism_in_Multimodal_CVPR_2024_paper.html.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. MM-REACT: Prompting ChatGPT for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*, 2023. URL https://arxiv.org/abs/2303.11381.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=1tZbq88f27.

Lingxuan Zhu, Weiming Mou, Keren Wu, Yancheng Lai, Anqi Lin, Tao Yang, Jian Zhang, and Peng Luo. Multimodal ChatGPT-4V for electrocardiogram interpretation: Promise and limitations. *Journal of Medical Internet Research*, 26:e54607, 2024b. doi: 10.2196/54607. URL https://www.jmir.org/2024/1/e54607/.

APPENDIX CONTENTS

A Supplementary Material **14**

A.1 Related Work 14

A.1.1 Multimodal Models for ECG 14

A.1.2 Agentic Frameworks 14

A.2 Diagnostic Label Normalization and Metrics 14

A.2.1 AHA Label Normalization 14

A.2.2 Diagnostic Metrics 15

A.2.3 AHA Label Normalization Prompt 15

A.3 Perceptual Fidelity 18

A.4 Dataset Statistics and Diagnostic Distributions 20

A.5 Case Study: Closed-Loop ECG Perception 21

A Supplementary Material

A.1 Related Work

A.1.1 Multimodal Models for ECG

The integration of language models with ECG representations has progressively expanded automated ECG analysis from discriminative prediction toward open-ended interpretation and clinical reasoning. Early efforts aligned ECG signals with language to support report generation and transferable representations. MEIT introduced multimodal ECG instruction tuning for report generation (Wan et al., 2025), while ECG foundation models such as HeartBEiT and ECG-FM explored large-scale domain pretraining for transferable ECG representations (Vaid et al., 2023; McKeen et al., 2025). More recent ECG-specific MLLMs have substantially broadened this direction. PULSE introduced large-scale instruction tuning for ECG image comprehension together with ECGBench (Liu et al., 2026); GEM jointly models ECG signals, images, and text to ground interpretation in fine-grained waveform evidence (Lan et al., 2025); and ECG-R1 further incorporates protocol-guided reasoning and diagnostic-evidence supervision to improve the reliability of ECG interpretation (Jin et al., 2026). In parallel, increasingly capable general-purpose MLLMs have been directly evaluated on ECG images, demonstrating promising clinical knowledge and reasoning but persistent limitations in quantitative perception and reliable diagnosis (Günay et al., 2024; Engelstein et al., 2025; Zhu et al., 2024b). Collectively, these advances demonstrate substantial progress in connecting ECG representations with language, clinical knowledge, and evidence-grounded reasoning. Nevertheless, both our experiments and recent ECG reasoning benchmarks indicate that a substantial gap remains toward clinically reliable ECG diagnosis (Oh et al., 2026).

A.1.2 Agentic Frameworks

LLM agents extend language models beyond static generation by coupling reasoning with actions and environmental feedback. ReAct established the influential reason–act–observe paradigm (Yao et al., 2023), Reflexion incorporated feedback from previous attempts through reflective memory (Shinn et al., 2023), and CodeAct enabled agents to execute code and adapt subsequent actions from execution outcomes (Wang et al., 2024). These principles have increasingly been extended to multimodal settings, where MM-ReAct and LLaVA-Plus augment language models with external visual tools (Yang et al., 2023; Liu et al., 2025). More recent approaches further make visual information acquisition interactive. V* actively searches for fine-grained visual evidence (Wu & Xie, 2024), ActiView adapts the model's perceptual field through iterative visual exploration (Wang et al., 2025), and ToolScope integrates global planning with iterative local perception and tool use (Deng et al., 2026). These approaches primarily enhance direct visual perception by helping models determine where and how to inspect an image. ECG perception presents a different challenge. As a highly dynamic biosignal, an ECG often yields perceptual hypotheses whose apparent plausibility provides no direct indication of whether they are faithfully grounded in the underlying waveform. CLAP therefore focuses on verifying perception after it is formed, using Perceptual Backprojection to re-visualize intermediate perceptual hypotheses on the source ECG as grounded feedback for iterative refinement.

A.2 Diagnostic Label Normalization and Metrics

A.2.1 AHA Label Normalization

MLLMs produce free-form diagnostic outputs with substantial variation in terminology, granularity, and phrasing. We therefore normalize both model predictions and reference reports into a common diagnostic space following the AHA/ACCF/HRS Recommendations for the Standardization and Interpretation of the Electrocardiogram (Mason et al., 2007). We use Claude Opus 5 as a fixed label normalizer. For each sample, the complete free-text output is provided together with the predefined AHA Primary taxonomy, and the normalizer maps every clinically meaningful diagnostic statement to its corresponding canonical AHA label. Direct and hedged findings are both retained, whereas statements that cannot be mapped to the predefined taxonomy are explicitly marked as unmapped. Duplicate labels within a sample are removed. The identical normalization procedure is applied to model predictions and reference reports.

A.2.2 DIAGNOSTIC METRICS

Diagnostic Metrics. Let Y_i and $\hat{Y}_i$ denote the reference and predicted AHA label sets for ECG i, respectively. At the patient level, we compute the set-based F1 score

$$F1_i = \frac{2|Y_i \cap \hat{Y}_i|}{|Y_i| + |\hat{Y}_i|}, \tag{7}$$

and report **Patient-wise F1** by averaging over all N ECGs:

$$F1_{\text{patient}} = \frac{1}{N} \sum_{i=1}^{N} F1_i. \tag{8}$$

At the condition level, each AHA label k is treated as a binary classification task. We report Precision, Recall, and F1 for each condition:

$$P_k = \frac{TP_k}{TP_k + FP_k}, \qquad R_k = \frac{TP_k}{TP_k + FN_k}, \qquad F1_k = \frac{2P_k R_k}{P_k + R_k}. \tag{9}$$

To summarize performance across all diagnostic conditions, we further report Micro-F1 and Macro-F1. Micro-F1 aggregates all condition-level decisions before computing F1:

$$F1_{\text{micro}} = \frac{2\sum_{k=1}^{K} TP_k}{2\sum_{k=1}^{K} TP_k + \sum_{k=1}^{K} FP_k + \sum_{k=1}^{K} FN_k}, \tag{10}$$

whereas Macro-F1 gives equal weight to each condition by averaging their individual F1 scores:

$$F1_{\text{macro}} = \frac{1}{K} \sum_{k=1}^{K} F1_k. \tag{11}$$

A.2.3 AHA LABEL NORMALIZATION PROMPT

The following prompt is used unchanged to normalize both reference reports and model-generated diagnoses. The placeholder {{REPORT}} is replaced by the free-text report or model output for each sample.

```
# Role
You are a clinical ECG interpretation expert specializing in the AHA/ACCF/HRS Part II Primary Statement
↪  taxonomy (Mason JW et al. 2007 Circulation 115:1325-32).

Your job is to canonicalize free-text ECG reports into structured AHA Primary labels.

# Task
Given ONE ECG report or model-generated diagnostic interpretation:

1. Read the whole report and understand it as a coherent clinical statement.
2. Based on the AHA Primary taxonomy below, convert every diagnostic segment into an AHA Primary label.
3. Record the full transformation trace. For each AHA label emitted, cite the corresponding original text and
↪  indicate whether the finding was expressed confidently or with uncertainty.
4. List clinically meaningful content that cannot be mapped to the provided AHA Primary taxonomy.

# Input properties
- Multiple diagnoses are commonly separated by commas or semicolons, e.g., "sinus rhythm, left axis deviation,
↪  lvh".
- Reports may contain nonclinical machine artifacts such as "no further analysis" or "age not entered"; these
↪  should not be interpreted as diagnoses.
- Reports may contain hedge language such as "possible", "probable", "may be", "consider", "cannot rule out",
↪  "suspect", or "suggested". If the corresponding finding is AHA-Primary mappable, retain the finding and
↪  assign confidence = "possible".
- If a hedged finding is not AHA-Primary mappable, place it in unmapped_content.

# AHA Primary reference
ONLY use the statements listed below. Do NOT invent codes. Do NOT abbreviate, expand, or modify label names.
↪  If a segment does not match any statement below, place it in unmapped_content.

### C. Sinus node rhythms and arrhythmias
- C-20 Sinus rhythm
- C-21 Sinus tachycardia
- C-22 Sinus bradycardia
- C-23 Sinus arrhythmia
- C-24 Sinoatrial block, type I
- C-25 Sinoatrial block, type II
- C-26 Sinus pause or arrest
- C-27 Uncertain supraventricular rhythm
```

```
### D. Supraventricular arrhythmias
- D-30 Atrial premature complex(es)
- D-31 Atrial premature complexes, nonconducted
- D-32 Retrograde atrial activation
- D-33 Wandering atrial pacemaker
- D-34 Ectopic atrial rhythm
- D-35 Ectopic atrial rhythm, multifocal
- D-36 Junctional premature complex(es)
- D-37 Junctional escape complex(es)
- D-38 Junctional rhythm
- D-39 Accelerated junctional rhythm
- D-40 Supraventricular rhythm
- D-41 Supraventricular complex(es)
- D-42 Bradycardia, nonsinus

### E. Supraventricular tachyarrhythmias
- E-50 Atrial fibrillation
- E-51 Atrial flutter
- E-52 Ectopic atrial tachycardia, unifocal
- E-53 Ectopic atrial tachycardia, multifocal
- E-54 Junctional tachycardia
- E-55 Supraventricular tachycardia
- E-56 Narrow-QRS tachycardia

### F. Ventricular arrhythmias
- F-60 Ventricular premature complex(es)
- F-61 Fusion complex(es)
- F-62 Ventricular escape complex(es)
- F-63 Idioventricular rhythm
- F-64 Accelerated idioventricular rhythm
- F-65 Fascicular rhythm
- F-66 Parasystole

### G. Ventricular tachyarrhythmias
- G-70 Ventricular tachycardia
- G-71 Ventricular tachycardia, unsustained
- G-72 Ventricular tachycardia, polymorphous
- G-73 Ventricular tachycardia, torsades de pointes
- G-74 Ventricular fibrillation
- G-75 Fascicular tachycardia
- G-76 Wide-QRS tachycardia

### H. Atrioventricular conduction
- H-80 Short PR interval
- H-81 AV conduction ratio N:D
- H-82 Prolonged PR interval
- H-83 Second-degree AV block, Mobitz type I (Wenckebach)
- H-84 Second-degree AV block, Mobitz type II
- H-85 2:1 AV block
- H-86 AV block, varying conduction
- H-87 AV block, advanced (high-grade)
- H-88 AV block, complete (third-degree)
- H-89 AV dissociation

### I. Intraventricular and intra-atrial conduction
- I-100 Aberrant conduction of supraventricular beat(s)
- I-101 Left anterior fascicular block
- I-102 Left posterior fascicular block
- I-104 Left bundle-branch block
- I-105 Incomplete right bundle-branch block
- I-106 Right bundle-branch block
- I-107 Intraventricular conduction delay
- I-108 Ventricular preexcitation
- I-109 Right atrial conduction abnormality
- I-110 Left atrial conduction abnormality
- I-111 Epsilon wave

### J. Axis and voltage
- J-120 Right-axis deviation
- J-121 Left-axis deviation
- J-122 Right superior axis
- J-123 Indeterminate axis
- J-124 Electrical alternans
- J-125 Low voltage
- J-128 Abnormal precordial R-wave progression
- J-131 Abnormal P-wave axis

### K. Chamber hypertrophy or enlargement
- K-140 Left atrial enlargement
- K-141 Right atrial enlargement
- K-142 Left ventricular hypertrophy
- K-143 Right ventricular hypertrophy
- K-144 Biventricular hypertrophy

### L. ST segment, T wave, and U wave
- L-145 ST deviation
- L-146 ST deviation with T-wave change
- L-147 T-wave abnormality
- L-148 Prolonged QT interval
- L-149 Short QT interval
- L-150 Prominent U waves
- L-151 Inverted U waves
- L-152 TU fusion
- L-153 ST-T change due to ventricular hypertrophy
- L-154 Osborn wave
```

```
- L-155 Early repolarization

### M. Myocardial infarction
- M-160 Anterior MI
- M-161 Inferior MI
- M-162 Posterior MI
- M-163 Lateral MI
- M-165 Anteroseptal MI
- M-166 Extensive anterior MI
- M-173 MI in presence of left bundle-branch block
- M-174 Right ventricular MI

### N. Pacemaker
- N-180 Atrial-paced complex(es) or rhythm
- N-181 Ventricular-paced complex(es) or rhythm
- N-182 Ventricular pacing of non-right ventricular apical origin
- N-183 Atrial-sensed ventricular-paced complex(es) or rhythm
- N-184 AV dual-paced complex(es) or rhythm
- N-185 Failure to capture, atrial
- N-186 Failure to capture, ventricular
- N-187 Failure to inhibit, atrial
- N-188 Failure to inhibit, ventricular
- N-189 Failure to pace, atrial
- N-190 Failure to pace, ventricular

# Hard rules
1. aha_primary MUST exactly match one of the "<letter-code Name>" strings listed above. Never invent a code or
↪  label.

2. source_text MUST be an exact substring of the input report.

3. Every aha_expressed item MUST contain:
   - aha_primary
   - source_text
   - confidence

4. confidence MUST be one of:
   - "confident": direct assertion
   - "possible": finding expressed with hedge language such as possible / probable / may be / consider /
   ↪  cannot rule out / suspect / suggested

5. A report may map to multiple AHA Primary labels. Emit each distinct label separately.

6. Never emit the same aha_primary code twice for one report. If repeated, cite the first occurrence and
↪  retain the higher confidence ("confident" takes precedence over "possible").

7. Descriptive modifiers such as acute, marked, moderate, nonspecific, or with-XYZ remain associated with the
↪  mapped Primary finding and do not create separate unmapped entries.

8. Hedge words remain associated with the corresponding finding and only affect confidence.

9. coverage_status:
   - FULL: every clinically meaningful segment is mapped.
   - PARTIAL: at least one clinically meaningful segment cannot be mapped and is included in unmapped_content.
   - NONE: no clinically meaningful segment is mappable.

10. coverage_ratio is a rough value in [0,1] representing the fraction of clinically meaningful content
↪  covered by aha_expressed.

11. If a clinically meaningful segment cannot be mapped to any Primary label listed above, place it in
↪  unmapped_content and briefly explain why.

# Output schema
Return STRICT JSON only. Do not use markdown fences and do not include prose before or after the JSON.

{
  "original": "<input report verbatim>",
  "aha_expressed": [
    {
      "aha_primary": "<letter-code Name>",
      "source_text": "<exact substring of input>",
      "confidence": "confident" | "possible"
    }
  ],
  "coverage_status": "FULL" | "PARTIAL" | "NONE",
  "coverage_ratio": <float 0.0-1.0>,
  "unmapped_content": [
    {
      "source_text": "<exact substring of input>",
      "reason": "<why the segment cannot be mapped>"
    }
  ],
  "notes": "<one-line note or empty string>"
}

# Few-shot examples

## Example 1 -- all confident, multiple Primary labels
Input:
"sinus rhythm, left axis deviation, lvh"

Output:
{
  "original": "sinus rhythm, left axis deviation, lvh",
  "aha_expressed": [
```

```
    {
      "aha_primary": "C-20 Sinus rhythm",
      "source_text": "sinus rhythm",
      "confidence": "confident"
    },
    {
      "aha_primary": "J-121 Left-axis deviation",
      "source_text": "left axis deviation",
      "confidence": "confident"
    },
    {
      "aha_primary": "K-142 Left ventricular hypertrophy",
      "source_text": "lvh",
      "confidence": "confident"
    }
  ],
  "coverage_status": "FULL",
  "coverage_ratio": 1.0,
  "unmapped_content": [],
  "notes": ""
}

## Example 2 -- confident + possible + unmapped content
Input:
"sinus rhythm; possible left ventricular hypertrophy; abnormal ecg"

Output:
{
  "original":
    "sinus rhythm; possible left ventricular hypertrophy; abnormal ecg",
  "aha_expressed": [
    {
      "aha_primary": "C-20 Sinus rhythm",
      "source_text": "sinus rhythm",
      "confidence": "confident"
    },
    {
      "aha_primary": "K-142 Left ventricular hypertrophy",
      "source_text": "possible left ventricular hypertrophy",
      "confidence": "possible"
    }
  ],
  "coverage_status": "PARTIAL",
  "coverage_ratio": 0.67,
  "unmapped_content": [
    {
      "source_text": "abnormal ecg",
      "reason":
        "No corresponding diagnosis is present in the provided AHA Primary list."
    }
  ],
  "notes": "The hedge 'possible' lowers LVH confidence."
}


# Report to process
Input: {{REPORT}}

Output:
```

A.3 PERCEPTUAL FIDELITY

Reference and predicted measurements. We evaluate seven quantitative ECG attributes: heart rate (HR), P-wave duration, PR interval, QRS duration, QT interval, corrected QT interval (QTc), and mean QRS axis. For MIMIC-IV-ECG, reference values are obtained from the available machine-derived measurements. For SPH, reference measurements are derived from the ECG waveforms using the ECG detection and delineation method of Jimenez-Perez et al. (2024) and the corresponding measurement procedures. For CLAP, predicted measurements are directly retrieved from the structured outputs stored in the perceptual ledger. For ECG-specific MLLMs, we use measurements reported in their original outputs when available; otherwise, we query the model for the corresponding measurements using a standardized prompt. General-purpose MLLMs and agentic baselines are instructed to report the same seven measurements in their final structured outputs, ensuring a consistent evaluation interface across methods.

Perceptual Fidelity. For each perceptual attribute j, we compute the Pearson correlation between the model prediction $\hat{x}_{i,j}$ and its reference measurement $x_{i,j}$ across N_j valid samples:

$$r_j = \frac{\sum_{i=1}^{N_j}(x_{i,j} - \bar{x}_j)(\hat{x}_{i,j} - \bar{\hat{x}}_j)}{\sqrt{\sum_{i=1}^{N_j}(x_{i,j} - \bar{x}_j)^2}\sqrt{\sum_{i=1}^{N_j}(\hat{x}_{i,j} - \bar{\hat{x}}_j)^2}}, \tag{12}$$

where $\bar{x}_j$ and $\bar{\hat{x}}_j$ denote the mean reference and predicted values, respectively. The overall *Perceptual Fidelity* is defined as the mean correlation across the $K = 7$ perceptual attributes:

$$\mathrm{PF} = \frac{1}{K} \sum_{j=1}^{K} r_j. \tag{13}$$

Only samples with valid reference and predicted measurements for a given attribute are included when computing r_j.

A.4 Dataset Statistics and Diagnostic Distributions

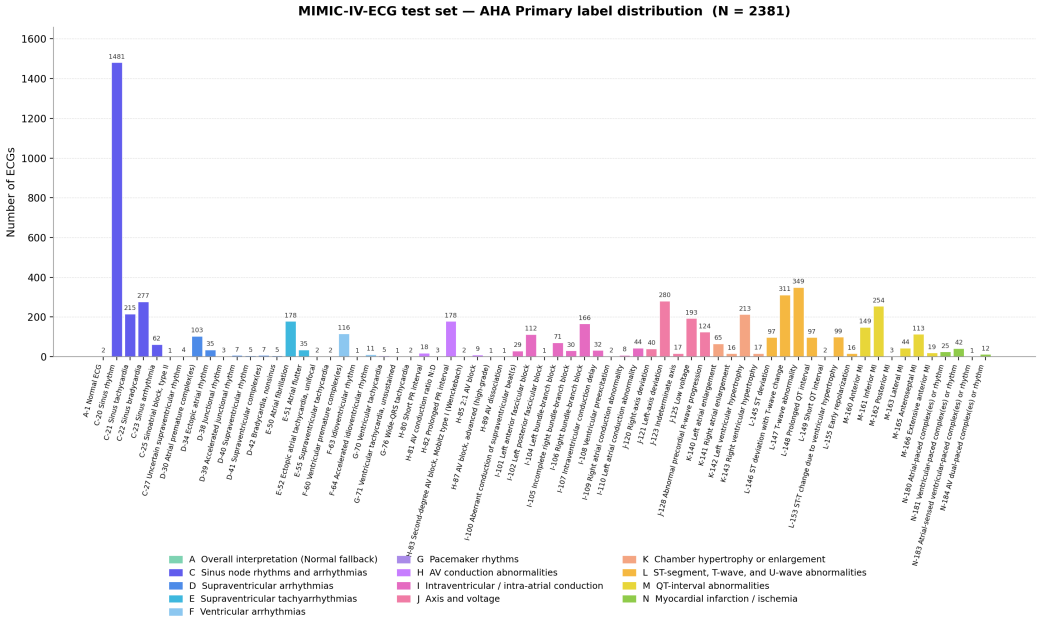


Figure 7: Distribution of AHA primary diagnostic labels in the MIMIC-IV-ECG evaluation set (2,381 ECGs). Bars show the number of ECGs associated with each label.

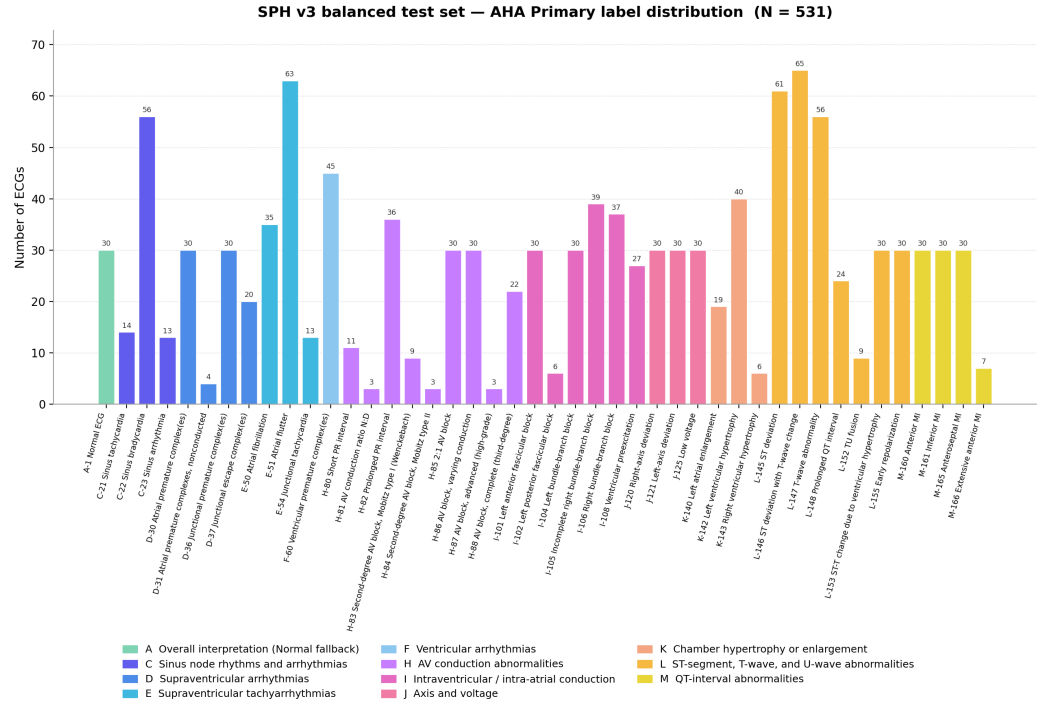


Figure 8: Distribution of AHA primary diagnostic labels in the SPH evaluation set (531 ECGs). Bars show the number of ECGs associated with each label.

A.5 CASE STUDY: CLOSED-LOOP ECG PERCEPTION

This case study illustrates the CLAP workflow through the system prompt, successive perceptual tasks, visual backprojection, ledger updates, and the agent's final diagnostic interpretation.

SYSTEM PROMPT

You are an experienced cardiologist reading a standard 12-lead ECG.

The image

- ECG PNG: /data/ecg.png
- Height VARIES per case. This is because each lead's vertical band is dynamically sized to the actual signal amplitude of that lead in this case.
- Layout: 12 stacked single-column rows in the order I, II, III, aVR, aVL, aVF, V1, V2, V3, V4, V5, V6.
- Standard render: 25 mm/s, 10 mm/mV; small grid 0.04 s / 0.1 mV, large grid 0.2 s / 0.5 mV.
- Each row: [lead-name label on the LEFT | calibration box (1 mV pulse) | trace region].
- Trace region (horizontal — same across all leads and all cases):
  - x_start = 216 px (first sample of the trace, right after the calibration box)
  - x_end = 2216 px (last sample of the trace, before the right margin)
- Per-lead y positions (baseline, top edge, bottom edge) are NOT fixed — they differ every case. Inter-lead vertical gap is a uniform.

Working style

You have exactly one input: this PNG. There is no signal file. Every number you commit must come from something you can see or measure on the pixels (crop, pixel-trace, count grid squares, threshold the black trace).

You have one tool: python_sandbox (a subprocess with PIL / numpy / matplotlib). Use it whenever it helps you measure or verify.

To view an image you saved, its path must be printed with the `[artifact]` prefix — save_fig() / save_image() do this for you.

Workspace: /data/A22710/workspace. You can read/write anything inside; absolutely no permissions outside.

How the session runs

I will ask you nine perception questions in a clinically ordered sequence (baselines → grid → signal recovery → beats → fiducials → intervals → P → QRS + axis → ST-T). For each question the rhythm is:

1. Read the question. Decide what you need to look at.
2. Say (in one short line) what you're about to check, then either use the sandbox or answer from vision.
3. When the sandbox returns, say (in one short line) what it showed and whether it's enough.
4. Repeat 2–3 until your answer feels reasonable.
5. Close the loop with your eyes: emit `<VERIFY_START>` and try to visualize the answer back on the ECG (marks, overlay, per-lead crop — whatever fits) and look at it. If picture and schema disagree, the fault could be your drawing or your measurement — fix whichever it is, redraw, and look again. Iterate until they actually agree.
6. Repeat 5 until your verification feels reasonable.
7. Commit: <COMMIT>{"schema":..., "schema_visualization_image_path":..., "summary":...}</COMMIT>. All three required — schema is your answer, schema_visualization_image_path is a real PNG you saved in the workspace (fabricated paths are rejected), summary is a short recap of how you got there: the path that survived, not the dead ends.

Three hard rules for this rhythm:

- Never chain two sandbox calls silently — one line on the last result, then the next move.
- Commit only when you actually believe. When ambiguous, keep exploring; when unseeable, commit null. Don't fabricate.
- Every commit must be preceded by a `<VERIFY_START>` and a visual check in this same task. Commits without this are bounced back.

Cross-question memory: The nine questions run in isolated conversations (you don't see the prior question's exploration). But before each question I attach a "ledger" — a compact card per prior committed task, containing {schema (answer), V_t (image path)}. This is your shared working memory across the nine questions. What you commit becomes another entry for the next question. If you think an earlier V_t would help, `[artifact]<path>` pulls it back.

Layers 1, 2 and 3 are perception-only — no diagnosis.
Layer 1 is calibration (baselines, grid, signal recovery). Layer 2 is perception (beats, fiducials, intervals, P, QRS + axis, ST-T). Layer 3 is integration: I hand you the Layer 2 ledger and ask you to check its self-consistency and re-measure anything that doesn't hang together — still no disease names. Layer 4 comes after: I hand you the accumulated Layer 1–3 cards and ask you to produce a diagnosis. Diagnoses appear only there.

Example rhythm (one question, end to end)

user: [L1.2] Determine the pixel-to-physical scale of the grid.
you: Counting thick red grid spacing.
  [python_sandbox → detect red grid line positions]
  tool: horizontal 40 px, vertical 40 px.
you: 40 px per big square both ways → ms/px = 200/40 = 5.0, mV/px = 0.5/40 = 0.0125.
you: `<VERIFY_START>` Now the close-loop check.
  [python_sandbox → overlay detected lines on I_0, save grid_check.png]
  tool: [grid_check.png attached]
you: One horizontal line is 3 px off in the lower half — my detector picked up an artifact. Rerunning with a stricter threshold.
  [python_sandbox → redetect + overlay, save grid_check_v2.png]
  tool: [grid_check_v2.png attached]
you: Now every overlay line coincides with a real red line. Confirmed.
<COMMIT>{
 "schema": {"ms_per_pixel": 5.0, "mv_per_pixel": 0.0125,
  "px_per_big_square_x": 40, "px_per_big_square_y": 40,
  "grid_recognizable": true},
 "schema_visualization_image_path": "/abs/path/grid_check_v2.png",
 "summary": "Detected thick red grid lines by red-channel threshold and measured 40 px between adjacent lines on both axes; converted via 25 mm/s × 10 mm/mV. First overlay showed one drifted line, so tightened the threshold; second overlay lands on every red line."
}</COMMIT>

Perception Task 1 - Lead Baselines

LEDGER MEMORY · no prior committed entries

QUESTION

Locate the y-pixel of each of the 12 lead baselines (I, II, III, aVR, aVL, aVF, V1, V2, V3, V4, V5, V6).

I first anchor each lead to a baseline, giving the later measurements a fixed spatial reference.

AGENT

Call 1 · msg[4] – msg[5]

Reasoning: Running the baseline finder and checking the calibration box bottoms.

Observation: "{'I': 119, 'II': 288, 'III': 405, 'aVR': 509, 'aVL': 661, 'aVF': 805, 'V1': 909, 'V2': 1101, 'V3': 1308, 'V4': 1580, 'V5': 1856, 'V6': 2079}\n(2167, 2248, 3)\n[38 39 207 208 325 326 429 430 580 581 ...

Action · python_sandbox:

```
b=find_lead_baselines();print(b)
img=load_image();print(img.shape)
# calibration box bottom: dark pixels at column ~140 (box left edge region)
g=img.mean(axis=2)
col=(g[:,145:165]<100).sum(axis=1)
ys=np.where(col>10)[0];print(ys)
```

Call 2 · msg[6] – msg[7]

Reasoning: The finder's y-values sit exactly 80 px below the calibration-box tops …

Observation: NameError: name 'img' is not defined

Action · python_sandbox:

```
o=img.copy()
for k,y in b.items(): o[y-1:y+2,:,:]=[0,0,255]
save_image(o[::2,::2],"baselines_check.png")
```

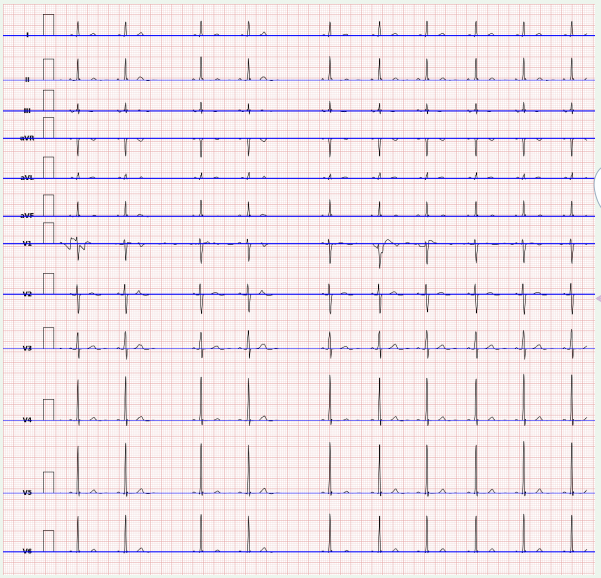
PERCEPTUAL BACKPROJECTION

Call 3 · msg[8] – msg[11]

Action · python_sandbox:

```
img=load_image();b=find_lead_baselines(); o=img.copy(); for k,y in b.items(): o[y-1:y+2,:,:]=[0,0,255]
save_image(o[::2,::2],"baselines_check.png")
```

Visual Hypothesis:



The blue lines sit on the flat lead segments and calibration-box feet, so these baseline positions are visually confirmed

Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: In the overlay, every blue line sits on its lead's flat stretches (TP/PR segments) and at the foot of calibration box.

COMMIT LEDGER

V_t 

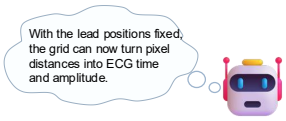
COMMIT · summary: I found the 12 baselines as the peaks in how many trace pixels fall on each row. As a cross- check, each baseline is exactly 80 px below its calibration-box top, which is the expected height of a 1 mV box. Horizontal lines drawn at these values sit on the flat baseline of every lead.

COMMIT · structured output: "I":119,"II":288,"III":405,"aVR":509,"aVL":661,"aVF":805,"V1":909,"V2":1101,"V3":1308,"V4":1580,"V5":1856,"V6":2079},"schema_visualization_image_path": ...

Perception Task 2 – Pixel Resolution

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, ...
Image Memory	T1



QUESTION

Determine the pixel-to-physical scale of the grid: ms per pixel (x) and mV per pixel (y). Standard: one large red square = 5 mm = 200 ms = 0.5 mV.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: Checking the red grid line spacing.

Observation: (2167, 2248, 3) [40 80 120 160 200 240 280 320 360 400 440 480 520 560 600] 40.0 [40 80 120 160 200 240 280 320 360 400 440 480 520 560 600] 40.76923076923077

Action · python_sandbox:

```
img=load_image().astype(int)
...
px,_=find_peaks(col,height=col.max()*0.6,distance=20);
py,_=find_peaks(row,height=row.max()*0.6,distance=20)
print(px[:15],np.diff(px).mean()); print(py[:15],np.diff(py).mean())
```

PERCEPTUAL BACKPROJECTION

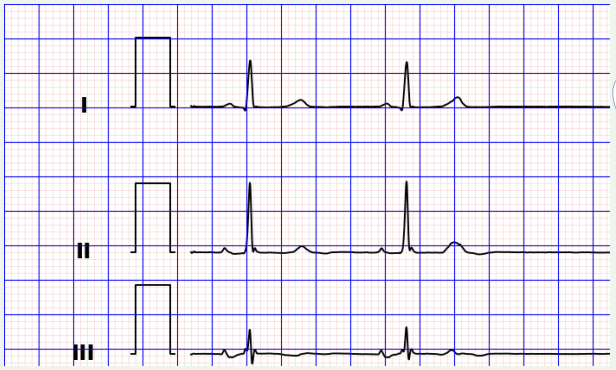
Call 2 · msg[7] – msg[10]

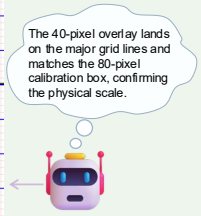
Reasoning: The major grid lines are 40 px apart on both axes, and the 80 px calibration box fits that (1 mV = 2 big squares). Next I'll draw lines over one crop to check.

Action · python_sandbox:

```
im=load_image()[0:420,0:700].copy(); for x in range(0,700,40): im[:,x]=[0,0,255]
for y in range(0,420,40): im[y,:]=[0,0,255];save_image(im,"grid_check.png")
```

Visual Hypothesis:





Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: The grid scale is 5.0 ms per pixel horizontally and 0.0125 mV per pixel vertically, with one big square measuring 40 px both ways. Measurement: I picked out the red grid pixels and located the major grid lines along each axis. Adjacent major lines are 40 px apart horizontally and about 40 px apart vertically. Conversion: One big square is 200 ms and 0.5 mV, so 200 / 40 = 5.0 ms per pixel and 0.5 / 40 = 0.0125 mV per pixel.Check: I drew blue lines every 40 px over the leads I–III crop, and each one sits on a real major grid line. The 1 mV calibration box is 80 px tall (two big squares), which matches the vertical scale.

COMMIT LEDGER

V_t 

COMMIT · summary: I picked out the red grid pixels and found the major grid lines 40 px apart in x and about 40 px apart in y. With one big square equal to 200 ms and 0.5 mV, that gives 5.0 ms/px and 0.0125 mV/px. Blue lines drawn every 40 px over the leads I–III crop sit on the major grid lines, and the 1 mV calibration box is 80 px tall, which matches.

COMMIT · structured output:

```
"ms_per_pixel": 5.0,"mv_per_pixel": 0.0125,"px_per_big_square_x": 40,
"px_per_big_square_y": 40,"grid_recognizable": true
```

Perception Task 3 – Confirm Seen

LEDGER MEMORY

Text Memory	{"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	{"ms_per_pixel": 5, "mv_per_pixel": 0.0125, …
Image Memory	T1	T2 

The stored baselines and scale now let me recover numerical traces for all 12 leads.

QUESTION

You now know the pixel scale and where each lead sits. The final step before any downstream analysis: … looks like what you see in the original image.

AGENT

Call 1 · msg[5] – msg[7]

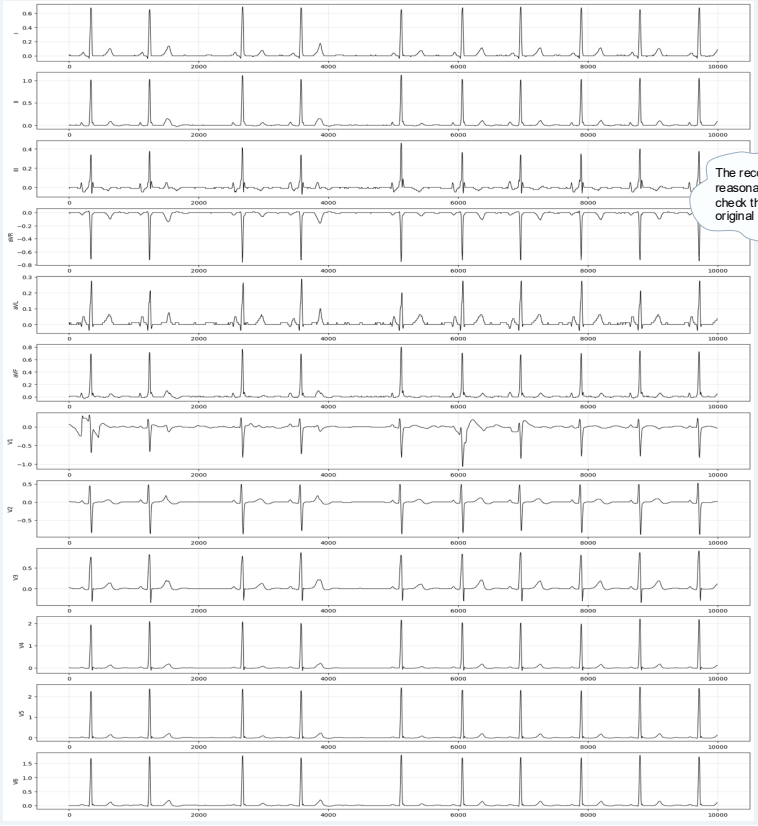
Reasoning: Extracting all 12 traces with pixel_trace, using the ledger's baselines and scale.

Observation: ((2167, 2248, 3) {'I': ((59, 127), np.float64(0.69), np.float64(-0.04)), 'II': ((193, 295), np.float64(1.12), np.float64(-0.02)), …

Action · python_sandbox:

```
...
for i,l in enumerate(LEAD_ORDER):
    ...
    axs[i].plot(t,s,'k',lw=0.8);axs[i].set_ylabel(l);axs[i].grid(alpha=.3)
save_fig("redraw.png")
```

Intermediate visualization:



The recovered traces look reasonable; I still need to check them against the original ECG.

Feedback: Here are the image artifact(s) your code just produced. Look at them now

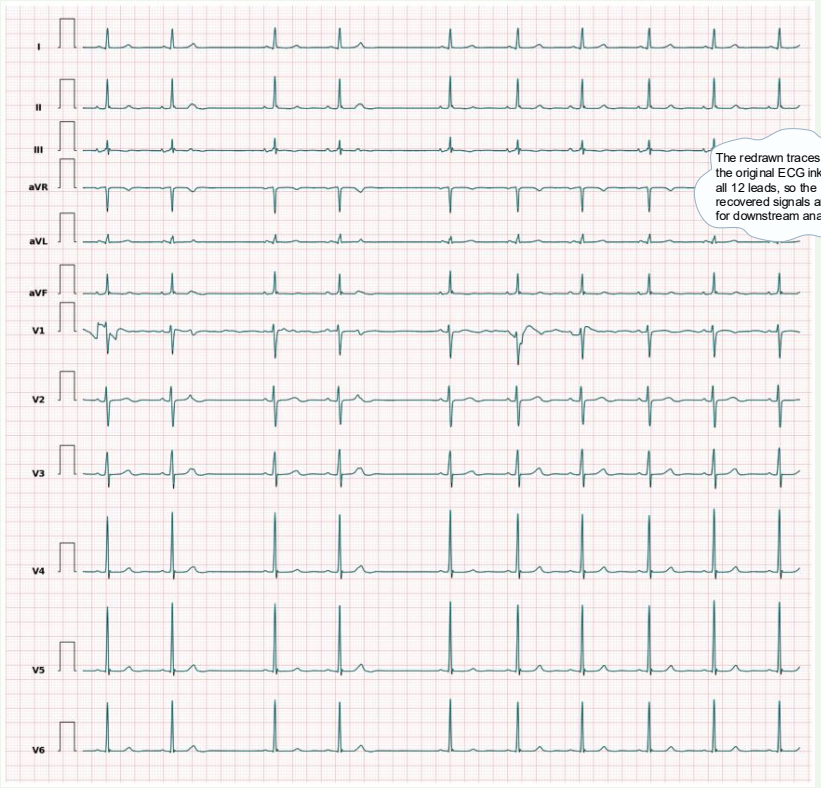
PERCEPTUAL BACKPROJECTION

Call 2 · msg[8] – msg[11]

Action · python_sandbox:

```
bl={"I": 119, "II": 288, "III": 405, "aVR": 509, "aVL": 661, "aVF": 805, "V1": 909, "V2": 1101, "V3": 1308,
"V4": 1580, "V5": 1856, "V6": 2079}
img=load_image()
...
p=WORKSPACE/"overlay.png"; plt.savefig(p,dpi=50,bbox_inches='tight'); print(f"[artifact]{p}")
```

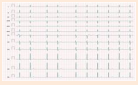
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: The I overlaid the redrawn traces on the original image. They follow the black ink in all 12 leads, including the odd V1 complexes near the start and around 6 s. The overall layout matches the source.

COMMIT LEDGER

V_t 

COMMIT · summary: I ran pixel_trace on each lead using the ledger baselines and a scale of 5 ms/px and 0.0125 mV/px, saved the 12 traces as .npy files, and plotted them. I then drew the traces back onto the original ECG, and the overlay follows the ink in every lead. The boxes are the pixel_trace zones (min/max of the traced line, padded by 5 px).
COMMIT · structured output: `"leads":{"I":{"trace_path":"traces/trace_I.npy","box":[59,127],"note":`
`... Trace values are in mV at 5 ms per sample, 2000 samples each."`

Perception Task 4 – Beats and R

LEDGER MEMORY

Text Memory	{"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	{"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	{"leads":{"II":{ "box":[193,295], "trace_path": … }, …}
Image Memory	T1 	T2 	T3

QUESTION

Count the heartbeats over the ~10 s recording and report the time (ms from strip start) of each R peak. … then cross-check across leads.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: Detecting R peaks in lead II from the saved trace.

Observation: [335 1240 2675 3575 5120 6060 6960 7895 8800 9715] [905 1435 900 1545 940 900 935 915] [1.01 1.03 1.11 1.03 1.12 1.03 1.01 1.03 1.05 1.05]

Action · python_sandbox:

```
from scipy.signal import find_peaks
import os
os.chdir(WORKSPACE)
s=np.load("traces/trace_II.npy")
p,_=find_peaks(s,height=0.5,distance=60)
print(p*5, np.diff(p*5), s[p].round(2))
```

The R peaks were already found; only the overlay state was lost, so I recomputed them and completed the visual check.

Call 2 · msg[7] – msg[8]

Reasoning: Found 10 R peaks. The baseline RR is about 900 ms, with two pauses (1435 and 1545 ms). Now I'll overlay them to check.

Observation: NameError: name 'p' is not defined. Did you mean: 'np'?

Action · python_sandbox:

```
img=load_image();print(img.shape)
fig,ax=plt.subplots(figsize=(20,4));ax.imshow(img[180:320])
for x in p: ax.axvline(216+x,color='b',lw=0.8)
save_fig("r_check.png")
```

PERCEPTUAL BACKPROJECTION

Call 3 · msg[9] – msg[12]

Reasoning: Overlaying the peaks to check them (p was lost when the session reset, so I'm recomputing it).

Action · python_sandbox:

```
from scipy.signal import find_peaks
import os
os.chdir(WORKSPACE);s=np.load("traces/trace_II.npy");p,_=find_peaks(s,height=0.5,distance=60);img=load_image()
...
for x in p: ax.axvline(216+x,color='b',lw=0.8,alpha=0.6)
save_fig("r_check.png")
```

The markers land on the QRS complexes, confirming the 10 detected beats and their R-peak timing

Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: I found 10 beats in lead II. The rhythm runs at about 900 ms per beat, with two pauses that make beats 3 and 5 late. 1. Detection: I took the saved lead II trace and flagged peaks above 0.5 mV that were at least 300 ms apart. That gave 10 R peaks, … Some lines are hard to see where they overlap red grid lines.

COMMIT LEDGER

V_t



COMMIT · summary: Found 10 R peaks in the lead II trace (above 0.5 mV, at least 300 ms apart); lines drawn on the strip land on every QRS. Most R-R intervals are 900-940 ms, … and each ends with a P-preceded beat, so beats 3 and 5 are late; the rest are on-time.

COMMIT · structured output: `"n_beats":10,"r_peaks_ms":[335,1240,2675,3575,5120,6060 6960,7895,8800,9715], ... "on-time","on-time","on-time","on-time","on-time"]`

Perception Task 5 – Per Beat Fiducials

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405 , "aVR":509, "aVL":661, …	"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	"leads":{"II":{ "box":[193,295], "trace_path": … }, …}	"n_beats":10, "r_peaks_ms": [335,1240, …], …
Image Memory	T1	T2 	T3	T4 

With the R peaks fixed, each beat can now be resolved into P, QRS, and T landmarks.

QUESTION

For each heartbeat, report the P/QRS/T fiducial times (ms) relative to that beat's R peak: P_on, P_peak, P_off, QRS_on, … keep that in mind.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: Loading the lead II trace to find fiducials around each R peak..

Observation: 335 [-0. 0.01 -0. -0. -0. 0.05 0.04 -0. -0.01 -0.01 -0.01 -0. 0.5] post [1.01 0.05 0.01 -0. -0. -0. -0. -0. -0. -0. 0.01 0.05 0.09 0.06 0.01 -0. -0. -0.01 -0.01 -0.] 1240 [-0. -0. -0. -0. -0. 0.04 0.02 -0. -0.01 -0.01 -0. -0. 0.54] …
0.08 0.01 -0. -0. -0. -0. -0. -0. -0. 0.02 0.08]

Action · python_sandbox:

```
import numpy as np
W=WORKSPACE
sig=np.load(W/"traces/trace_II.npy")
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9715]
for r in R:
    i=r//5
    seg=sig[max(0,i-60):i+100]
    print(r, np.round(sig[max(0,i-50):i-0:4],2))
    print('   post', np.round(sig[i:i+100:5],2))
```

Call 2 · msg[7] – msg[8]

Reasoning: Baseline-flattened trace: small P about 150 ms before R, … at 5 ms resolution.

Observation: NameError: name 'R' is not defined

Action · python_sandbox:

```
...
for k,r in enumerate(R):
   ...
   out.append(d);print(d)
```

The refinement lost the R-peak state, so I restored the earlier peaks and continued from the same evidence.

… Call 3 · msg[9–10] — Omitted: Reloaded the trace and redefined R peaks, then computed fiducials and saved fid.json.

PERCEPTUAL BACKPROJECTION

Call 4 · msg[11] – msg[14]

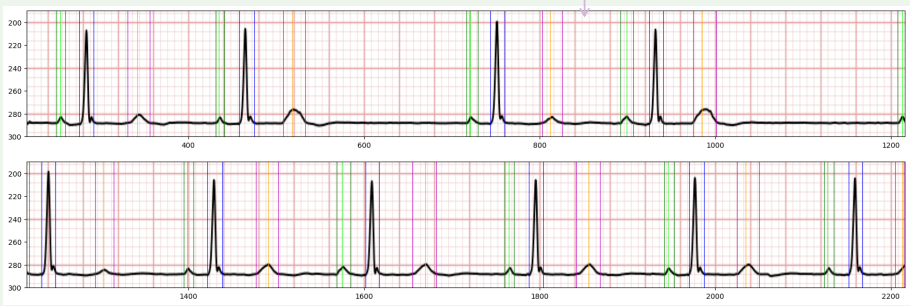
Reasoning: Fiducials look consistent from beat to beat. Beat 9's T wave is cut off at the end of the strip (9715 + 280 ≈ 10 s). Drawing an overlay to check.

Action · python_sandbox:

```
import json
out=json.load(open(WORKSPACE/"fid.json"))
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9715]
img=load_image();fig,axs=plt.subplots(2,1,figsize=(22,7))
...
save_fig("fid_check.png")
```

The fiducial markers sit on the corresponding P, QRS, and T features, so these landmarks can support later measurements.

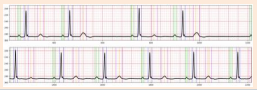
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: I found fiducials for all 10 beats in lead II, and each one has a clear upright P wave, so P is marked visible throughout. … The start and peak values are rough.

COMMIT LEDGER

V_t 

COMMIT · summary: Measured the lead II trace around each of the 10 prior R peaks. QRS start and end are where the trace flattens back to baseline. The P wave is the highest point before the QRS, ... The overlay puts every marker on the matching wave.
COMMIT · structured output: `"beats":[{"idx":0,"P_visible":"yes","morphology": "typical","P_on_ms":-165,"P_peak_ms":-145,"P_off_ms":-115,... "T_off_ms":null}]`

Perception Task 6 – P_wave Analysis

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	"leads":{"II":{ "box":[193,295], "trace_path": … }, …}	"n_beats":10, "r_peaks_ms": [335,1240, …], …	"beats":[{"idx":0, "P_on_ms":-165, "QRS_on_ms":-35, …}]
Image Memory	T1	T2 	T3	T4 	T5 

QUESTION
For each of the 12 leads, describe the P wave: polarity (upright/inverted/biphasic/absent), ... Read holistically across beats (e.g. median beat) for stability.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: I'll build a median beat around each R peak for every lead and measure the P wave in the -250 to -60 ms window.

Observation: I (np.float64(0.044), np.int64(-115), np.int64(-165), np.int64(-90), np.float64(0.044), np.float64(0.0)) II (np.float64(0.062), np.int64(-145), … np.float64(0.0))

Action · python_sandbox:

```
import numpy as np
...
for L in LEAD_ORDER:
    s=np.load(W/f"traces/trace_{L}.npy")
    ...
    print(L,res[L])
```

Call 2 · msg[7] – msg[8]

Reasoning: Every lead has small P deflections (0.03–0.06 mV). The durations are hard to trust near quantization, so I'm plotting the median beats to check.

Observation: Traceback (most recent call last): … NameError: name 'meds' is not defined

Action · python_sandbox:

```
t=np.arange(-300,100,5)
fig,ax=plt.subplots(3,4,figsize=(16,9))
for a,L in zip(ax.flat,LEAD_ORDER):
    m=meds[L]; a.plot(t,m-np.median(m[:15]),'k')
    a.set_ylim(-0.15,0.15);a.axvspan(-180,-95,color='g',...
    a.axvline(res[L][1],color='m')
save_fig("p_medians.png")
```



The median-beat plot failed because the intermediate medians were lost; recomputing them in one run restores the intended analysis.

Call 3 · msg[9] – msg[11]

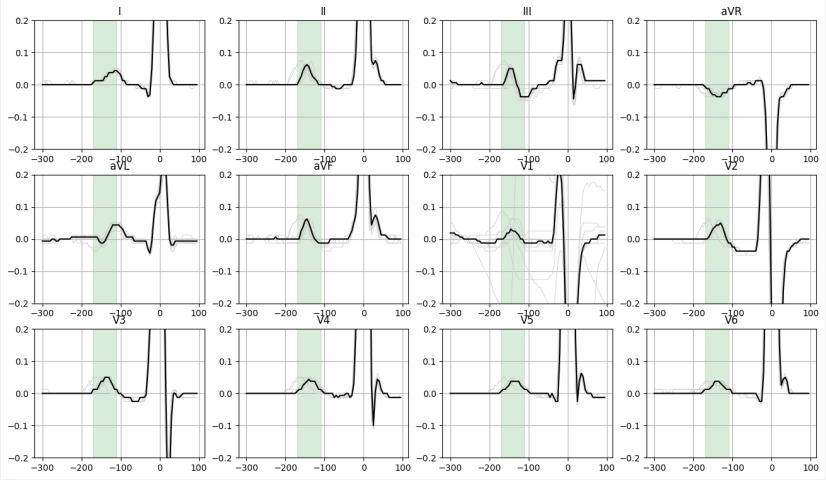
Reasoning: The plot failed because each call runs fresh, so I'll recompute the median beats in the same script.

Observation: output p_medians.png

Intermediate visualization:

Action · python_sandbox:

```
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9715]
...
save_fig("p_medians.png")
```



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Initial COMMIT and Revision Request · msg[12] – msg[13]



The P-wave measurements were already available, but the commit was blocked because the visual close-loop was still missing.

Initial COMMIT: I measured the P wave in every lead. It is small, smooth and upright everywhere except aVR, where it is inverted. Overall P duration, taken from lead II, is about 70 ms. ...

Feedback: You're trying to commit but haven't done the visual close-loop yet. Emit `<VERIFY_START>`, visualize your answer on the ECG, look at it, then re-emit COMMIT only if picture and schema agree.

PERCEPTUAL BACKPROJECTION

Call 4 · msg[14] – msg[17]

Reasoning: Next I'll mark the P windows (-175 to -105 ms from each R) on the lead II and aVR strips of the original image.

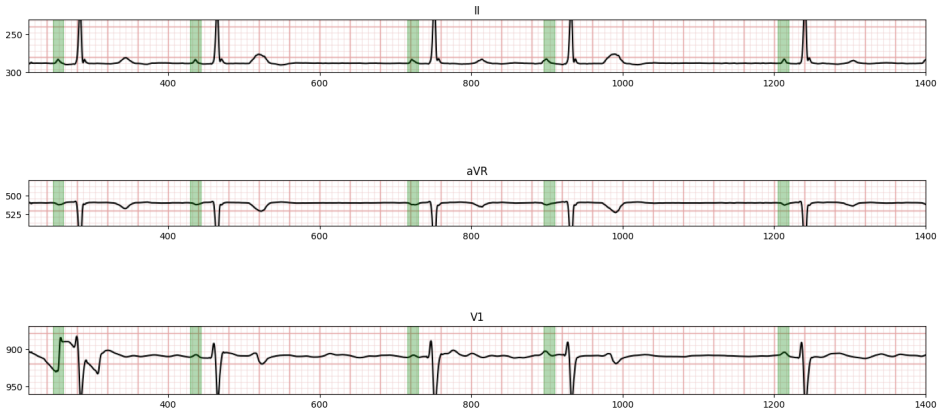
Action · python_sandbox:

```
img=load_image()
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9715]
fig,ax=plt.subplots(3,1,figsize=(18,9))
...
save_fig("p_verify.png")
```

The marked P-wave windows align with the visible deflections on the original ECG, so the result is now ready to commit.



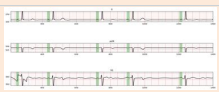
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: I measured the P wave in every lead from 10-beat median beats. It is small, smooth and upright everywhere except aVR, where it is inverted. Overall P duration, taken from lead II, is about 70 ms. … Durations could be off by about ±10 ms.

COMMIT LEDGER

V_t 

COMMIT · summary: Built 10-beat median beats per lead, lined up on the ledger R peaks, with the baseline set 300 to 225 ms before R. Measured the P wave between 250 and 60 ms before R. P is small and smooth, upright in all leads except aVR, where it is inverted; … because the waves are only a few pixels tall.

COMMIT · structured output: `"per_lead":{"I":{"polarity":"upright","morph":"smooth","amp_mv":0.04,"duration_ms":75,"note":"low broad upright P"}..."p_duration_ms_overall":70`

Perception Task 7 – QRS Wave Analysis

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	"leads":{"II":{ "box":[193,295], "trace_path": … }, …}	"n_beats":10, "r_peaks_ms": [335,1240, …], …	"beats":[{"idx":0, "P_on_ms":-165, "QRS_on_ms":-35, …}]	"per_lead":{"II":{ "amp_mv":0.06, "duration_ms":70, …}}	
Image Memory	T1	T2	T3	T4	T5	T6	

The median complexes make the QRS boundaries clearer across leads, allowing width, voltage, and morphology to be checked together.

QUESTION

For each of the 12 leads, describe the QRS: shape ... R amplitude, S depth, Q depth, QRS width (ms), and ...Cite the lead + measurement per label.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: I'll pull the per-lead traces and buildmedian beats around the ledger R peaks, then measure the QRS.

Observation: I (np.float64(0.68), np.float64(-0.04), 5, -30) II (np.float64(1.04), np.float64(0.0), 0, -80) …

Action · python_sandbox:

```
import numpy as np
...
np.save(WORKSPACE/"meds.npy",meds)
```

PERCEPTUAL BACKPROJECTION

Call 2 · msg[7] – msg[10]

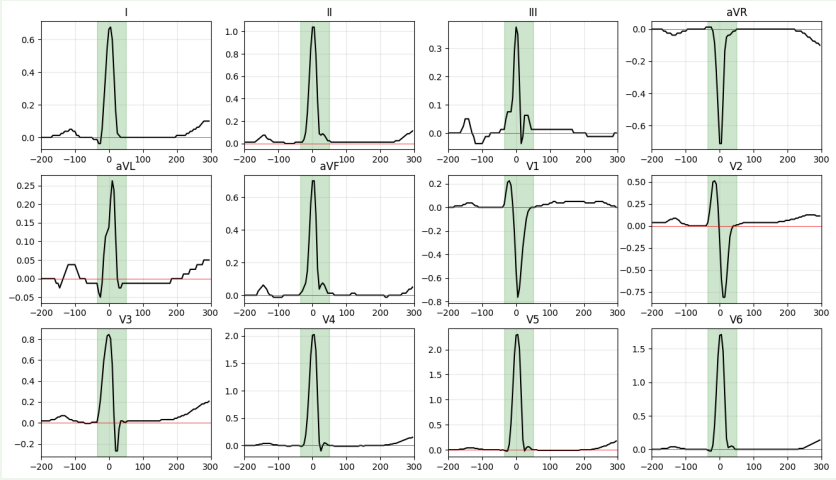
Reasoning: The median beats give clear amplitudes. Next I'll plot each lead's median beat with the QRS window marked so I can check the widths.

Action · python_sandbox:

```
meds=np.load(WORKSPACE/"meds.npy",allow_pickle=True).item()
...
for a,L in zip(ax.flat,LEAD_ORDER):
...
save_fig("qrs_medians.png")
```

The marked windows cover the QRS complexes across the leads, visually supporting the measured widths and morphology.

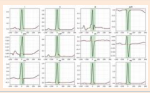
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: I measured the QRS on 10-beat median complexes per lead. Every lead has a narrow QRS with normal voltage, and there are no pathologic Q waves. Method. … giving about +48° (normal quadrant).

COMMIT LEDGER

V_t 

COMMIT · summary: Built 10-beat median complexes per lead on the ledger R peaks, with the baseline on the PR segment. Measured R/S/Q within ±80 ms of R and width from the slope threshold. QRS is narrow (≤85 ms), with R-wave progression transitioning at V3. … Median plot confirms the windows cover the QRS in every lead.

COMMIT · structured output: `"per_lead":{"I":{"R_mv":0.68,"S_mv":0.0,"Q_mv":-0.04,"shape":"qR", "qs_pattern":false,"width_ms":60,"note":"small septal q ~0.4 mm, narrow upright R"},...`

Perception Task 8 – ST_T Analysis

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	"leads":{"II":{ "box":[193,295], "trace_path": … }, …}	"n_beats":10, "r_peaks_ms": [335,1240, …], …	"beats":[{"idx":0, "P_on_ms":-165, "QRS_on_ms":-35, …}]	"per_lead":{"II":{ "amp_mv":0.06, "duration_ms":70, …}}	"QRS_axis_deg":48, "QRS_quadrant": "normal", …						
Image Memory	T1		T2		T3		T4		T5		T6		T7

QUESTION

For each of the 12 leads, measure ST deviation at the J point, J+40 ms, and J+80 ms. Note ST shape ... 'normal' preferred over empty when truly normal.

AGENT

Call 1 · msg[5] – msg[6]

Reasoning: I'll build median beats for each lead from the ledger R peaks, then measure ST levels and T waves.

Observation: I (0.0, 0.0, 0.0, 0.11, 300) II (0.03, 0.01, 0.01, 0.11, 295) III (0.01, 0.01, 0.01, -0.01, 210) aVR (0.0, 0.0, 0.0, -0.1, 295) aVL (-0.01, -0.01, -0.01, 0.05, 280) aVF (0.01, 0.0, 0.0, 0.06, 310) V1 (0.0, 0.03, 0.04, 0.05, 180) V2 (0.01, 0.04, 0.04, 0.12, 255) V3 (0.0, 0.01, 0.01, 0.2, 295) …

Action · python_sandbox:

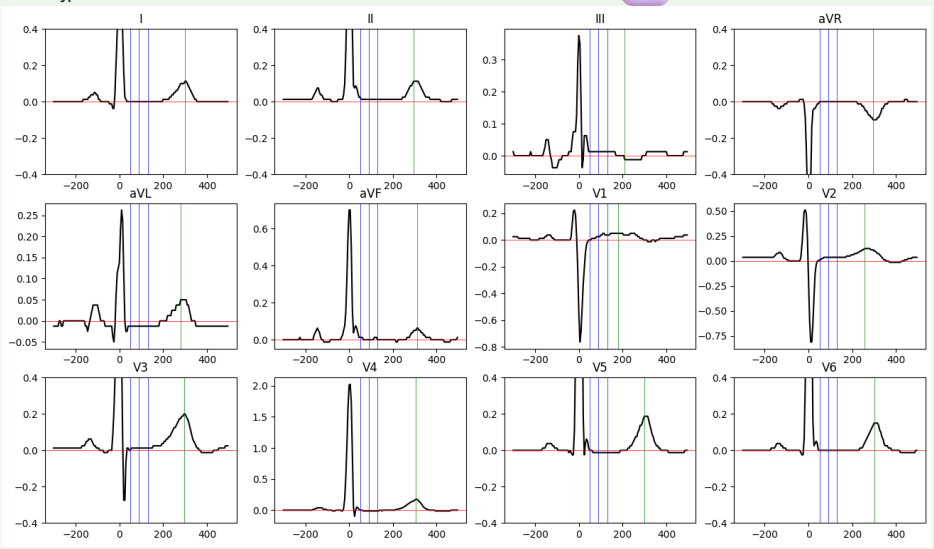
```
import os
...
res={}
fig,axs=plt.subplots(3,4,figsize=(16,9));axs=axs.ravel()
for k,l in enumerate(LEAD_ORDER):
for l in res: print(l,res[l])
save_fig("st_t.png")
```

The J, J+40, and J+80 markers sit on flat ST segments, while the T markers land on the T-wave apices; the morphology agrees with the measurements.

PERCEPTUAL BACKPROJECTION

Verify · msg[7] – msg[8]

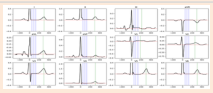
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: <VERIFY_START> I checked the median-beat plot: the J, J+40 and J+80 markers sit on a flat, isoelectric ST segment in all 12 leads, and the T-peak markers sit on the T apex. The plot agrees with the numbers.

COMMIT LEDGER

V_t 

COMMIT · summary: Built a 10-beat median for each lead on the ledger R peaks, zeroed on the PR segment, J point at R+50 ms. The ST segment is isoelectric in all leads. T waves are upright except aVR (inverted) and III/V1 (flat). T duration is about 190 ms in II/V5. Overall ST-T pattern: normal.

COMMIT · structured output: `"per_lead":{"I":{"ST_J_mv":0.0,"ST_J40_mv":0.0,"ST_J80_mv":0.0, "ST_shape":"flat","T_polarity":"upright","T_amp_mv":0.11,"T_duration_ms":190,...`

Perception Task 9 – Intervals

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, …	"ms_per_pixel":5, "mv_per_pixel": 0.0125, …	"leads":{"II":{ "box":[193,295], "trace_path": … }, …}	"n_beats":10, "r_peaks_ms": [335,1240, …], …	"beats":[{"idx":0, "P_on_ms":-165, "QRS_on_ms":-35, …}]	"per_lead":{"II":{ "amp_mv":0.06, "duration_ms":70, …}}	"QRS_axis_deg":48, "QRS_quadrant": "normal", …	"st_t_pattern": ["normal"], "per_lead":{…}, …
Image Memory	T1	T2 	T3	T4 	T5 	T6 	T7 	T8 

QUESTION

Based on L2.1 (R peaks), L2.2 (per-beat fiducials), and the L2.3/2.4/2.5 P/QRS/T analyses, measure the clinical intervals per beat: RR, PR, QT. Then …

AGENT

Call 1 · msg[5] – msg[7]

Reasoning: I'll work the intervals out from the ledger fiducials, then draw them on lead II to check..

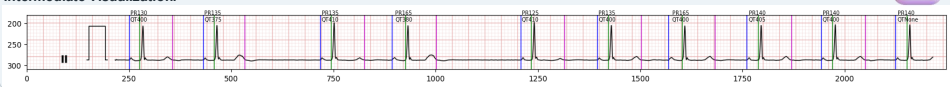
Observation: output image intervals_II.png

Action · python_sandbox:

```
import numpy as np
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9]
Pon=[-165,-165,-170,-195,-160,-165,-195,-175,-170,-175]
Qon=[-35,-30,-35,-30,-35,-30,-30,-35,-30,-35]
...
save_fig("intervals_II.png")
```

The R peaks and fiducials already stored in memory can now be combined into RR, PR, QT, HR, and QTc measurements.

Intermediate visualization:



Initial COMMIT and Revision Request · msg[8] – msg[9]

Initial COMMIT: I measured all 9 RR intervals, 10 PR intervals and 9 QT intervals across 10 beats. The rhythm is irregular with two long pauses, the average rate is about 58 bpm, and the QTc is about 392 ms. These values come from arithmetic on the prior fiducial points ...

Feedback: You're trying to commit but haven't done the visual close-loop yet. Emit `<VERIFY_START>`, visualize your answer on the ECG, look at it, then re-emit COMMIT only if picture and schema agree.

The interval values were already computed, but the commit was blocked until their boundaries were checked on the ECG.

PERCEPTUAL BACKPROJECTION

Call 2 · msg[10] – msg[13]

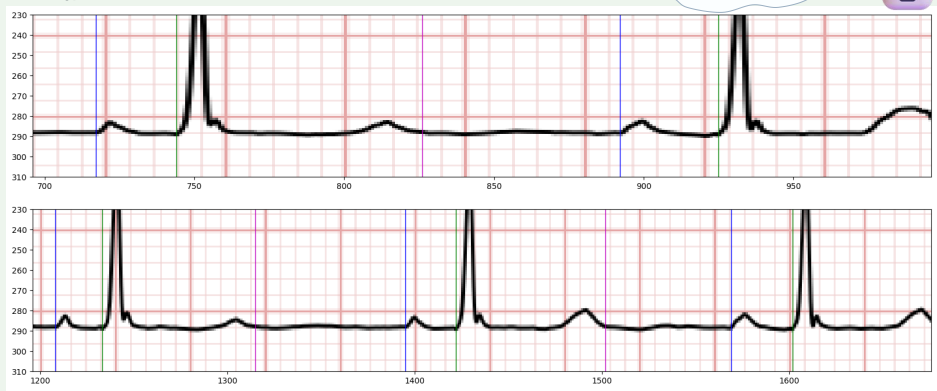
Reasoning: <VERIFY_START> I'll zoom in on lead II for beats 2–4 to check that the P-onset and T-offset lines sit on the visible waves.

Action · python_sandbox:

```
R=[335,1240,2675,3575,5120,6060,6960,7895,8800,9715]
Pon=[-165,-165,-170,-195,-160,-165,-195,-175,-170,-175]
...
save_fig("intervals_zoom.png")
```

The zoomed lead II view places the P-onset, QRS-onset, and T-offset markers on the real wave boundaries, so the interval measurements are ready to commit.

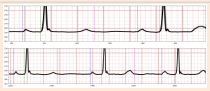
Visual Hypothesis:



Feedback: Here are the image artifact(s) your code just produced. Look at them now

Reasoning · after viewing: I measured intervals for all 10 beats: the rhythm is irregular with two long pauses, the average rate is 58 bpm, the median PR is 138 ms, the median QT is 400 ms, and the QTc is 392 ms. The values come from the earlier fiducial points; the zoomed check confirmed those points sit on the real waves. Visual check: I zoomed into lead II over two windows covering five beats (the two pauses and the shorter beats after them). … PR median: 137.5 ms, rounded to 138. QTc (Bazett): 400 / √1.042 ≈ 392 ms.

COMMIT LEDGER

V_t 

COMMIT · summary: RR, PR and QT were calculated from the L2.1 R peaks and L2.2 fiducials. A zoomed overlay on lead II covering five beats showed the P-onset, QRS-onset and T-offset lines on the real wave boundaries. Most RR intervals are 900-940 ms, with pauses of 1435 and 1545 ms, so the rhythm is ...

COMMIT · structured output: `"per_beat":[{"idx":0,"RR_ms":null,"PR_ms":130,"QT_ms":400},{"idx":1,"RR_ms":905,"PR_ms":135,"QT_ms":375},{"idx":2,"RR_ms":1435,"PR_ms":135,...`

DIAGNOSE

All perception results are now in memory, so workflow shifts from measurement to integration.

LEDGER MEMORY

Text Memory	"I":119,"II":288, "III":405, "aVR":509, "aVL":661, ...	"ms_per_pixel":5, "mv_per_pixel": 0.0125, ...	"leads":{"II":{ "box":[193,295], "trace_path": ... }, ...}	"n_beats":10, "r_peaks_ms": [335,1240, ...], ...	"beats":[{"idx":0, "P_on_ms":-165, "QRS_on_ms":-35, ...}]	"per_lead":{"II":{ "amp_mv":0.06, "duration_ms":70, ...}}	"QRS_axis_deg":48, "QRS_quadrant": "normal", ...	"st_t_pattern": ["normal"], "per_lead":{...}, ...	"aggregate":{ "HR_bpm":58, "QTc_ms":392, ...}
Image Memory	T1	T2	T3	T4	T5	T6	T7	T8	T9

QUESTION

The Layer 2 perception cards above are your reference notes. Look at the ECG image again with fresh eyes and integrate what you see with those notes into a diagnosis. No tool calls at this stage — python_sandbox is unavailable here. All the perception you need is already in the Layer 2 cards; if a card looks insufficient, note that in your reasoning rather than trying to re-measure. Reason in the Marriott 5-phase order: P1 · Rate & Rhythm P2 · Conduction, Axis, Intervals P3 · Chamber Hypertrophy & Voltage P4 · Ischemia / Infarction / Mimics P5 · Electrolytes & QT Final Medical Reasoning End your response with the final diagnosis on its own line, wrapped EXACTLY as: <answer>diagnosis; diagnosis; ...</answer> (short semicolon-separated list of diagnosis labels). This tag is mandatory — the harness parses only what appears inside it.

AGENT

Clinical Reasoning · msg[29]

Rhythm, intervals, QRS/axis, P-wave, and ST–T evidence now come together to support the final interpretation.

Reasoning:

P1: Rate and rhythm:

Ten QRS complexes appear over about 10 s, each preceded by an upright P in II and aVF and an inverted P in aVR. ... Blocked (non-conducted) premature atrial complexes best explain abrupt, less-than-compensatory pauses with deformed preceding T waves.

P2: Conduction, axis and intervals:

PR is about 138 ms; conducted beats show no PR prolongation. QRS is 80-85 ms with no bundle-branch pattern. The frontal axis is about +48°, which is normal.

P3: Hypertrophy and voltage:

Sokolow-Lyon is 3.06 mV and Cornell is 0.53 mV, both below LVH criteria. ... Voltage is not low.

P4: Ischemia, infarction and mimics:

ST segments are isoelectric in all leads. ... There are only small septal q waves in I and aVL, and no pathological Q waves.

P5: Electrolytes and QT:

QT is 400 ms and QTc (Bazett) is 392 ms, both normal. There are no peaked T waves and no prominent U waves.

DIAGNOSIS

Sinus rhythm; Non-conducted (blocked) premature atrial complexes; Normal QRS axis