PPG2ECG: Physiology-Guided Latent Rectified Flow for Electro-Hemodynamic PPG-to-ECG Generation

Xiaoda Wang[1,3]*, **Minxiao Wang**[2], **Kaiqiao Han**[3], **Defu Cao**[4], **Ching Chang**[3], **Yidan Shi**[3], **Runze Yan**[2], **Xiao Luo**[5], **Yan Liu**[4], **Xiao Hu**[2], **Yizhou Sun**[3], **Wei Wang**[3], **Carl Yang**[1]

[1]Department of Computer Science, Emory University
[2]Nell Hodgson Woodruff School of Nursing, Emory University
[3]Department of Computer Science, University of California, Los Angeles
[4]Department of Computer Science, University of Southern California
[5]Department of Statistics, University of Wisconsin–Madison

Code: https://github.com/ignite-abd/PPG2ECG

Abstract

Electrocardiography (ECG) is the clinical standard for cardiac assessment but requires dedicated hardware that does not scale to daily-life monitoring. Photoplethysmography (PPG) is ubiquitous in wearables but lacks ECG-specific diagnostic morphology and is corrupted by motion and sensor noise. PPG-to-ECG generation aims to bridge this gap by recovering electrical morphology and timing from peripheral pulse signals. However, existing methods largely rely on statistical alignment and data-driven generation. They fail to explicitly structure the latent space around physiology-aware electro-hemodynamic factors and lack constraints from forward physiological dynamics. To address these challenges, we propose **PPG2ECG**, a physiology-guided latent rectified flow framework. PPG2ECG introduces an electro-hemodynamic simulator that co-models ECG and PPG through shared cardiac phase dynamics. Guided by this simulator, a Physiology-Aware AutoEncoder learns a structured electro-hemodynamic latent space. Then we integrate this simulator guidance into a PPG-conditioned latent rectified flow, enforcing ECG-side morphology consistency and ECG-to-PPG forward hemodynamic consistency during generative transport. Experiments on the large-scale MC-MED dataset demonstrate that PPG2ECG significantly improves PPG-to-ECG generation and downstream cardiovascular disease classification, proving its ability to generate ECGs that are both signal-faithful and physiologically plausible under the ECG-to-PPG hemodynamic pathway.

1 Introduction

Electrocardiography (ECG) is the clinical standard for measuring cardiac electrical activity and diagnosing cardiovascular conditions [27, 55, 26, 54, 23, 24]. However, continuous ECG monitoring requires dedicated electrodes and stable skin contact, limiting its scalability in daily-life settings [53]. Photoplethysmography (PPG), in contrast, is widely available in consumer wearables, offering a low-cost measurement of peripheral blood-volume changes [1, 57, 5, 68, 39, 65]. Despite its appeal, PPG is fundamentally limited by high levels of motion artifacts and the inherent absence of diagnostic electrical features like P-QRS-T morphology [16, 4, 7]. As a delayed hemodynamic response, raw PPG is structurally deficient for direct and reliable clinical diagnosis. Overcoming this bottleneck by generating diagnostically informative ECGs from wearable PPG therefore provides a highly meaningful bridge between ubiquitous sensing and verifiable cardiac assessment [45, 52, 47, 56].

*Correspondence to: Xiaoda Wang <xiaoda.wang@emory.edu>.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Recent PPG-to-ECG studies have progressed from supervised waveform reconstruction to adversarial, diffusion-based, token-based, and latent flow-based generation [52, 45, 47, 10, 15]. Despite these advances, most existing methods treat PPG-to-ECG generation primarily as a data-driven cross-modal translation problem, leaving two key challenges: (i) *Lack of physiology-aware latent abstraction.* PPG and ECG are coupled but asymmetric observations of the same cardiovascular process: ECG reflects upstream electrical activation, whereas PPG captures the delayed peripheral pulse [2]. Purely statistical alignment fails to explicitly encode this directional physiological coupling. (ii) *Missing physiological simulator guidance.* Existing methods optimize data-driven signal similarity and easily hallucinate physically impossible hemodynamic delays, ignoring cardiac-cycle dynamics and the forward ECG-to-PPG hemodynamic pathway [52]. Incorporating this knowledge is non-trivial; naively imposing a fixed simulator as a hard constraint risks severe model mismatch and over-regularization. A robust approach must instead integrate simulator knowledge as soft, phase-aware guidance while retaining the flexibility of data-driven latent generation.

To address these challenges, we propose PPG2ECG, a physiology-guided latent rectified flow framework for electro-hemodynamic PPG-to-ECG generation. We introduce a novel electro-hemodynamic simulator that explicitly co-models ECG and PPG by linking an upstream electrical readout with a delayed peripheral pulse via shared cardiac phase dynamics. Using this simulator as a physiological anchor, PPG2ECG employs a Physiology-Aware AutoEncoder (PAE) to abstract modality-specific observations into a structured cardiovascular latent space regularized by cross-modal alignment and latency-aware consistency. On top of this latent space, PPG2ECG introduces a simulator-guided latent rectified flow. During training, the generative transport is explicitly regularized by the simulator to ensure both ECG-side morphological fidelity and forward ECG-to-PPG hemodynamic consistency. In summary, our main contributions are as follows:

❶ ***Physiology-Guided Generative Formulation.*** We introduce PPG2ECG, formulating PPG-to-ECG generation as an electro-hemodynamic generation problem. We design an electro-hemodynamic physiological simulator to co-model ECG and PPG through shared cardiac phase dynamics, upstream electrical readouts, and delayed hemodynamic readouts.

❷ ***Physiology-Aware Representation Learning.*** We design a Physiology-Aware AutoEncoder that abstracts modality-specific observations into a structured electro-hemodynamic latent space, rigorously constrained by simulator-derived latency-aware consistency, and cross-modal latent alignment.

❸ ***Simulator-Guided Flow Matching.*** We integrate the electro-hemodynamic simulator into PPG-conditioned latent rectified flow. Simulator-derived constraints guide the model to generate ECGs that are waveform-faithful and physiologically plausible under the ECG-to-PPG hemodynamic pathway.

❹ ***Multi-Level Experimental Validation.*** We demonstrate that PPG2ECG consistently improves PPG-to-ECG generation across waveform fidelity, clinical interval consistency, and cardiovascular disease detection on the large-scale MC-MED dataset.

2 Related Work

PPG-to-ECG Generation. PPG-to-ECG generation aims to recover cardiac electrical waveforms from PPG. One line of work treated this as supervised signal reconstruction, utilizing CNNs, RNNs, or Transformers [28] and specialized architectures like W-Net [52]. To enhance waveform realism, another line of work adopted generative models. For example, CardioGAN [45] and CLEP-GAN [31] employed adversarial and contrastive learning to improve subject-independent translation. More recently, diffusion and flow-based models have driven state-of-the-art high-fidelity translation. RDDM [47] utilizes a region-disentangled diffusion process for targeted ECG components, while CardioPPG [10] generates ECGs through a tokenized frequency-domain pathway. Most closely related to our work, PPGFlowECG [15] employs a two-stage latent rectified flow, generating ECGs from a shared cross-modal latent space. Despite these advancements, existing models rely predominantly on statistical data-driven alignment, lacking explicit regularization from forward physiological dynamics.

Physiological Simulators. Compact mathematical models utilize low-dimensional differential equations to simulate the cardiac cycle and generate stereotyped cardiovascular waveforms [37, 11]. The canonical ECGSYN model [37], for instance, employs a three-dimensional limit-cycle oscillator with phase-locked Gaussian components to synthesize realistic P–QRS–T morphologies and heart

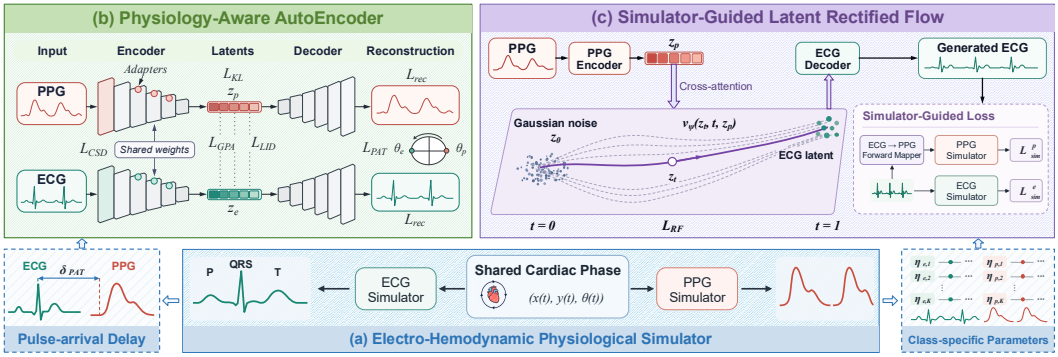

Figure 1: **Overview Framework of PPG2ECG.** (a) *The Electro-Hemodynamic Physiological Simulator*: co-models ECG and PPG through shared cardiac phase dynamics. (b) *Physiology-Aware AutoEncoder*: learns an aligned electro-hemodynamic latent space from resampled PPG and ECG windows. (c) *Simulator-Guided Latent Rectified Flow*: generates ECG latents conditioned on PPG and regularizes the transport with ECG morphology and ECG-to-PPG hemodynamic consistency.

rate variability. Similar oscillator-based frameworks have since been developed to model delayed peripheral hemodynamics, such as PPG [51]. However, purely mechanistic simulators rely on globally fixed templates, limiting their expressivity in capturing subject-specific conduction abnormalities or complex cross-modal variations. To overcome this, recent hybrid approaches integrate physiological priors with Neural ODEs [6, 58, 35] or Universal Differential Equations [43]. They can enable models to preserve essential physiological structures while flexibly fitting real-world clinical data [18, 17, 56].

Latent Generative Models. Diffusion and score-based generative models have become powerful tools for learning complex data distributions by gradually transforming noise into structured samples [21, 9, 67, 42, 32]. To improve computational efficiency, latent diffusion models move the generative process from the raw data space to a compressed latent space, reducing the cost of modeling high-dimensional signals while preserving sample quality [44]. This idea is particularly relevant for physiological waveform generation, where long, high-frequency, and noisy sensor sequences make direct generative modeling challenging [48]. Flow-based generative models provide an alternative continuous-time view of generation. Flow matching learns vector fields that transport simple base distributions to data distributions without simulating stochastic diffusion during training [33, 50, 60]. Rectified flow further encourages trajectories to follow straight transport paths between source and target distributions, enabling efficient sampling with coarse numerical integration [34, 30, 12]. These properties make rectified flow attractive for PPG-to-ECG generation, where the model must transport PPG-conditioned latent variables toward plausible ECG representations. Existing latent flow methods for PPG-to-ECG generation have shown strong empirical performance [15], but their transport objectives are still largely statistical.

Cross-Modal Physiological Representation Learning. Learning aligned representations across physiological modalities is central to translating PPG into ECG. Since PPG and ECG are generated by different but coupled physiological processes, successful cross-modal learning must balance modality-specific feature extraction with shared cardiovascular abstraction. Prior methods often encourage alignment by using shared encoders, contrastive objectives, or paired latent reconstruction [39, 8, 13, 66, 49, 19]. CardioPPG aligns PPG and ECG embeddings through cross-modal contrastive learning before using the aligned PPG encoder for downstream cardiovascular prediction and ECG synthesis [10]. PPGFlowECG similarly builds a shared latent space through a CardioAlign encoder and performs ECG generation in that aligned space [15]. These approaches, together with latent cross-modal synthesis in other clinical modalities [14, 61], demonstrate that latent alignment can improve downstream utility and generation quality.

3 Method: PPG2ECG

We present PPG2ECG, a physiology-guided latent rectified flow framework for electro-hemodynamic PPG-to-ECG generation. PPG2ECG has three components. First, an electro-hemodynamic physiological simulator co-models ECG and PPG through shared cardiac phase dynamics, an upstream

ECG readout, and a delayed PPG readout. Second, a Physiology-Aware AutoEncoder (PAE) learns a modality-aware shared latent space regularized by cross-modal alignment and simulator-derived latency consistency. Third, a PPG-conditioned latent rectified flow generates ECG latents from Gaussian noise, while simulator-derived residuals guide the transport toward ECG-side morphology consistency and ECG-to-PPG forward hemodynamic consistency. The simulator and the ECG-to-PPG forward mapper are used only during training; inference requires only the input PPG.

Problem Formulation. Given a paired PPG segment $\mathbf{x}_p \in \mathbb{R}^{L_p \times 1}$ and ECG segment $\mathbf{x}_e \in \mathbb{R}^{L_e \times 1}$, where L_p and L_e denote the numbers of PPG and ECG samples after resampling, our goal is to learn a conditional generator $p(\mathbf{x}_e \mid \mathbf{x}_p)$ that synthesizes a plausible ECG waveform consistent with the observed PPG. The Physiology-Aware AutoEncoder maps both modalities into aligned latent representations $\mathbf{z}_p, \mathbf{z}_e \in \mathbb{R}^{C_z \times T_z}$, where C_z and T_z denote the latent channel dimension and temporal length. The Electro-Hemodynamic Simulator Guided Latent Rectified Flow then learns a conditional vector field $\mathbf{v}_\psi(\mathbf{z}_t, t, \mathbf{z}_p)$ that transports a Gaussian latent variable toward the ECG latent distribution conditioned on $\mathbf{z}_p$. At inference time, only $\mathbf{x}_p$ is required: we encode it into $\mathbf{z}_p$, solve the learned flow to obtain $\hat{\mathbf{z}}_e$, and decode it into the generated ECG $\hat{\mathbf{x}}_e$.

3.1 The Electro-Hemodynamic Physiological Simulator

Because PPG is a delayed hemodynamic response rather than a direct electrical measurement, effective PPG-to-ECG generation requires a physiological anchor that captures the asymmetric ECG-to-PPG coupling. We therefore introduce an electro-hemodynamic physiological simulator composed of four ODE functions. The functions f_x and f_y define a shared cardiac phase oscillator, f_e generates the ECG voltage readout $e(t)$, and f_p generates the PPG pulse readout $p(t)$. The oscillator state (x, y) is not a measured signal; it acts as a cardiac clock whose polar angle θ is the phase shared by both readouts, so that ECG and PPG events are locked to the same beat. This simulator is not used as a rigid waveform template at inference time. Instead, it provides a soft training-time interface: the pulse-arrival delay δ_{PAT} regularizes the PAE latent space, while the vector fields f_e and f_p define finite-difference residuals for simulator-guided latent flow training. Following the ECGSYN dynamical model [37] and its use as a simulator prior for ECG generation [56], the shared cardiac phase follows a limit-cycle oscillator:

$$\frac{dx}{dt} = f_x(x, y) = \alpha(x, y)x - \omega y, \qquad \frac{dy}{dt} = f_y(x, y) = \alpha(x, y)y + \omega x, \tag{1}$$

where $\alpha(x, y) = 1 - \sqrt{x^2 + y^2}$ drives (x, y) toward the unit limit cycle, and $\theta(x, y) = \operatorname{atan2}(y, x) \in (-\pi, \pi]$ denotes the cardiac phase. In polar coordinates (r, θ), the radius obeys $\dot{r} = (1 - r)r$ while the phase advances at the constant rate $\dot{\theta} = \omega$. Hence one revolution of (x, y), with period $2\pi/\omega$, corresponds to one cardiac cycle, and ω sets the average heart rate.

The ECG readout follows the phase-locked Gaussian P–QRS–T formulation of ECGSYN [37]:

$$\frac{de}{dt} = f_e(x, y, e, t; \eta_e) = -\sum_{\beta \in \mathcal{B}_e} a^e_\beta \Delta\theta^e_\beta(x, y) \exp\left(-\frac{(\Delta\theta^e_\beta(x, y))^2}{2(b^e_\beta)^2}\right) - [e - e_0(t)], \tag{2}$$

where $\mathcal{B}_e = \{P, Q, R, S, T\}$, $\Delta\theta^e_\beta(x, y) = \left(\theta(x, y) - \theta^e_\beta\right) \bmod 2\pi$, and $e_0(t)$ is a slowly varying baseline (e.g., respiratory drift) toward which the readout relaxes. The parameters η_e collect the phase locations, amplitudes, and temporal widths of the ECG components. Analogous to Gaussian-kernel PPG pulse models [51], the PPG readout is driven by the same cardiac phase but delayed relative to the ECG R-peak:

$$\frac{dp}{dt} = f_p(x, y, p, t; \eta_p) = -\sum_{\gamma \in \mathcal{B}_p} a^p_\gamma \Delta\theta^p_\gamma(x, y) \exp\left(-\frac{(\Delta\theta^p_\gamma(x, y))^2}{2(b^p_\gamma)^2}\right) - \nu_p [p - p_0(t)], \tag{3}$$

where $\mathcal{B}_p = \{\mathrm{foot}, \mathrm{sys}, \mathrm{notch}, \mathrm{dia}\}$ denotes the p-foot, systolic peak, dicrotic notch, and diastolic wave, $p_0(t)$ is the PPG baseline, and $\nu_p > 0$ is the vascular relaxation rate toward it. The phase offset of each PPG component is defined as

$$\Delta\theta^p_\gamma(x, y) = \left(\theta(x, y) - \theta^e_R - \delta_{\mathrm{PAT}} - \theta^p_\gamma\right) \bmod 2\pi, \tag{4}$$

where θ_R^e is the ECG R-peak phase and δ_{PAT} denotes the pulse-arrival delay from electrical activation to the peripheral pulse response. The parameters η_p collect δ_{PAT}, the relaxation coefficient ν_p, and the phase, amplitude, and width parameters of the PPG components.

Euler Residual Interface. We discretize the simulator on modality-specific time grids using explicit Euler updates. For modality $m \in \{e, p\}$, let $\tilde{\mathbf{h}}_m$ denote a generated beat crop, let $\mathbf{s}_{m,\ell} = (x_{m,\ell}, y_{m,\ell})$ denote the corresponding reference cardiac phase state, and let f_m be the simulator vector field with fitted parameters η_m. The simulator induces the finite-difference residual

$$\mathbf{r}_\ell^m = \frac{\tilde{h}_{m,\ell+1} - \tilde{h}_{m,\ell}}{\Delta t_m} - f_m(\mathbf{s}_{m,\ell}, \tilde{h}_{m,\ell}, t_{m,\ell}; \eta_m). \tag{5}$$

Rather than enforcing the simulator as a hard waveform template, PPG2ECG uses this residual as soft physiological guidance during training. The full Euler discretization and simulator fitting procedure are provided in Appendix A.1.

3.2 Physiology-Aware AutoEncoder

The Physiology-Aware AutoEncoder (PAE) learns a shared cardiovascular latent space for paired PPG and ECG signals while preserving modality-specific acquisition characteristics. As shown in Figure 1, the encoder adopts a modality-specific-to-shared design. PPG and ECG are first processed by modality-specific blocks that preserve acquisition-dependent characteristics, such as waveform morphology, frequency content, and noise patterns. The resulting representations are then passed through shared blocks to extract common cardiovascular factors. To avoid losing useful modality-dependent information during shared abstraction, we insert lightweight modality-aware adapters in the intermediate shared blocks. A final shared bottleneck maps both modalities to a common latent posterior space. For modality $m \in \{p, e\}$, let $\mathbf{h}_m^{(0)} = \mathbf{x}_m$. The modality-specific blocks are defined as

$$\mathbf{h}_m^{(i)} = F_{m,i}^{\mathrm{spec}}(\mathbf{h}_m^{(i-1)}), \qquad i = 1, 2, \tag{6}$$

where $F_{m,i}^{\mathrm{spec}}$ denotes the i-th modality-specific block for modality m. The intermediate shared blocks with modality-aware adapters are then written as

$$\mathbf{u}_m^{(k)} = F_k^{\mathrm{sh}}(\mathbf{h}_m^{(k-1)}), \qquad \mathbf{h}_m^{(k)} = \mathbf{u}_m^{(k)} + A_{m,k}(\mathbf{u}_m^{(k)}), \qquad k = 3, \ldots, 5, \tag{7}$$

where F_k^{sh} is the k-th shared block and $A_{m,k}$ is the modality-aware adapter. The final shared bottleneck is

$$\mathbf{h}_m^{(6)} = F_6^{\mathrm{sh}}(\mathbf{h}_m^{(5)}). \tag{8}$$

The encoder parameterizes a diagonal Gaussian posterior by

$$[\boldsymbol{\mu}_m, \log \boldsymbol{\sigma}_m^2] = H_\phi(\mathbf{h}_m^{(6)}), \qquad q_\phi^m(\mathbf{z}_m \mid \mathbf{x}_m) = \mathcal{N}\left(\boldsymbol{\mu}_m, \mathrm{diag}(\boldsymbol{\sigma}_m^2)\right), \tag{9}$$

and samples the latent representation as

$$\mathbf{z}_m = \boldsymbol{\mu}_m + \boldsymbol{\sigma}_m \odot \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{10}$$

Here $\boldsymbol{\mu}_m, \log \boldsymbol{\sigma}_m^2 \in \mathbb{R}^{C_z \times T_z}$, so both PPG and ECG are mapped into aligned latent representations $\mathbf{z}_p, \mathbf{z}_e \in \mathbb{R}^{C_z \times T_z}$. The decoder for each modality reconstructs the waveform with the same number of samples as its input. It contains a residual reconstruction branch and a lightweight decomposition branch:

$$\hat{\mathbf{x}}_m = D_\theta^m(\mathbf{z}_m) = D_{\mathrm{res}}^m(\mathbf{z}_m) + D_{\mathrm{dec}}^m(\mathbf{z}_m), \tag{11}$$

where

$$D_{\mathrm{dec}}^m(\mathbf{z}_m) = D_{\mathrm{level}}^m(\mathbf{z}_m) + D_{\mathrm{trend}}^m(\mathbf{z}_m) + D_{\mathrm{season}}^m(\mathbf{z}_m). \tag{12}$$

The residual branch models local waveform morphology, while the decomposition branch provides a lightweight bias for global level, slow temporal drift, and coarse recurrent structure.

PAE Training Objective. PAE is optimized with three groups of objectives: waveform reconstruction at each modality's input rate, cross-modal latent alignment, and simulator-informed delay consistency.

(i) *Simulator-informed phase delay consistency.* PPG is a delayed hemodynamic response to ECG activation. Therefore, the phase gap between the generated ECG and PPG should be consistent with the class-specific simulator delay δ_{PAT}. Let $\hat{\mathbf{x}}_e = D_\theta^e(\mathbf{z}_e)$, and $\hat{\mathbf{x}}_p = D_\theta^p(\mathbf{z}_p)$, where $\hat{\mathbf{x}}_e$

and $\hat{\mathbf{x}}_p$ are the reconstructed ECG and PPG waveforms. Let $\Theta_e(\cdot)$ estimate the ECG activation phase and $\Theta_p(\cdot)$ estimate the PPG pulse phase. The estimated electro-hemodynamic phase delay is $\widehat{\Delta\theta}_{\mathrm{PAT}} = \mathrm{wrap}_\pi\left(\Theta_p(\hat{\mathbf{x}}_p) - \Theta_e(\hat{\mathbf{x}}_e)\right)$, where $\mathrm{wrap}_\pi(\cdot)$ maps the phase difference to $[-\pi, \pi]$. Since phase is circular, we use a circular squared distance: $d_{\mathrm{circ}}(a,b) = \left[\mathrm{atan2}\left(\sin(a-b), \cos(a-b)\right)\right]^2$. The phase delay consistency loss is then

$$\mathcal{L}_{\mathrm{PAT}} = \frac{1}{B}\sum_{i=1}^{B} d_{\mathrm{circ}}\left(\widehat{\Delta\theta}_{\mathrm{PAT}}^{(i)}, \delta_{\mathrm{PAT}}^{(i)}\right). \tag{13}$$

(ii) *Autoencoding regularization.* The reconstruction loss preserves modality-specific waveform information:

$$\mathcal{L}_{\mathrm{rec}} = \|D_\theta^p(\mathbf{z}_p) - \mathbf{x}_p\|_2^2 + \|D_\theta^e(\mathbf{z}_e) - \mathbf{x}_e\|_2^2. \tag{14}$$

The KL term regularizes both modality posteriors toward a standard Gaussian prior:

$$\mathcal{L}_{\mathrm{KL}} = D_{\mathrm{KL}}\left(q_\phi^p(\mathbf{z}_p \mid \mathbf{x}_p) \,\|\, \mathcal{N}(\mathbf{0}, \mathbf{I})\right) + D_{\mathrm{KL}}\left(q_\phi^e(\mathbf{z}_e \mid \mathbf{x}_e) \,\|\, \mathcal{N}(\mathbf{0}, \mathbf{I})\right). \tag{15}$$

(iii) *Cross-modal latent alignment.* We align paired PPG and ECG representations at both distribution and instance levels. Global posterior alignment matches posterior means and dispersion:

$$\mathcal{L}_{\mathrm{GPA}} = \left\|\boldsymbol{\mu}_e - \boldsymbol{\mu}_p\right\|_2^2 + \frac{1}{2}\left[D_{\mathrm{KL}}\left(q_\phi^e(\mathbf{z}_e \mid \mathbf{x}_e) \,\|\, q_\phi^p(\mathbf{z}_p \mid \mathbf{x}_p)\right) + D_{\mathrm{KL}}\left(q_\phi^p(\mathbf{z}_p \mid \mathbf{x}_p) \,\|\, q_\phi^e(\mathbf{z}_e \mid \mathbf{x}_e)\right)\right]. \tag{16}$$

To preserve instance-level distinctions, we use a bidirectional contrastive loss. Let $\bar{\mathbf{z}}_m^{(i)}$ denote the normalized pooled latent of the i-th sample from modality m, and define $s(\mathbf{a}, \mathbf{b}) = \exp(\langle \mathbf{a}, \mathbf{b}\rangle/\tau)$.

$$\mathcal{L}_{\mathrm{LID}} = -\frac{1}{2B}\sum_{i=1}^{B}\left[\log\frac{s(\bar{\mathbf{z}}_p^{(i)}, \bar{\mathbf{z}}_e^{(i)})}{\sum_{j=1}^{B} s(\bar{\mathbf{z}}_p^{(i)}, \bar{\mathbf{z}}_e^{(j)})} + \log\frac{s(\bar{\mathbf{z}}_e^{(i)}, \bar{\mathbf{z}}_p^{(i)})}{\sum_{j=1}^{B} s(\bar{\mathbf{z}}_e^{(i)}, \bar{\mathbf{z}}_p^{(j)})}\right]. \tag{17}$$

Finally, we use a weak cross-modal decodability objective to encourage each latent to retain modality-shared physiological information. And this objective is assigned a small weight and is intended only as a weak physiological decodability constraint.

$$\mathcal{L}_{\mathrm{CSD}} = \|D_\theta^e(\mathbf{z}_p) - \mathbf{x}_e\|_2^2 + \|D_\theta^p(\mathbf{z}_e) - \mathbf{x}_p\|_2^2. \tag{18}$$

The full PAE objective is

$$\mathcal{L}_{\mathrm{PAE}} = \lambda_{\mathrm{PAT}}\mathcal{L}_{\mathrm{PAT}} + \mathcal{L}_{\mathrm{rec}} + \beta\mathcal{L}_{\mathrm{KL}} + \lambda_{\mathrm{GPA}}\mathcal{L}_{\mathrm{GPA}} + \lambda_{\mathrm{LID}}\mathcal{L}_{\mathrm{LID}} + \lambda_{\mathrm{CSD}}\mathcal{L}_{\mathrm{CSD}}. \tag{19}$$

After PAE training, the PPG encoder, ECG encoder, and ECG decoder are frozen for the subsequent PPG-conditioned latent rectified flow.

3.3 Electro-Hemodynamic Simulator Guided Latent Rectified Flow

After PAE training, we freeze the PPG encoder, ECG encoder, and ECG decoder. For each paired sample, we obtain the PPG condition latent $\mathbf{z}_p = E_\phi^p(\mathbf{x}_p)$ and the target ECG latent $\mathbf{z}_e = E_\phi^e(\mathbf{x}_e)$. Following rectified flow [34] and its latent PPG-to-ECG variant [15], we learn a conditional vector field that transports Gaussian noise toward the ECG latent distribution conditioned on $\mathbf{z}_p$. Given $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $t \sim \mathcal{U}(0,1)$, we define the linear interpolation path

$$\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\mathbf{z}_e, \tag{20}$$

and optimize the flow-matching objective

$$\mathcal{L}_{\mathrm{RF}} = \mathbb{E}\left[\|\mathbf{v}_\psi(\mathbf{z}_t, t, \mathbf{z}_p) - (\mathbf{z}_e - \mathbf{z}_0)\|_2^2\right]. \tag{21}$$

The vector field $\mathbf{v}_\psi$ is parameterized by a conditional Transformer, where the evolving ECG latent attends to the PPG condition latent through cross-attention. To further constrain the generated ECG with electro-hemodynamic dynamics, we apply simulator guidance only to the terminal generated state rather than to every intermediate flow state. This avoids unstable gradients from early noisy latent states and aligns the physiological regularization with the final ECG waveform used at inference.

Given a noise latent z_0, we obtain a differentiable terminal estimate by integrating the learned vector field with K Euler steps:

$$\bar{z}_{k+1} = \bar{z}_k + \Delta\tau\, v_\psi(\bar{z}_k, \tau_k, z_p), \quad \tau_k = \frac{k}{K}, \quad \Delta\tau = \frac{1}{K}, \qquad \tilde{z}_e = \bar{z}_K, \quad \tilde{x}_e = D^e_\theta(\tilde{z}_e). \tag{22}$$

The decoded terminal ECG $\tilde{x}_e$ is then used to compute simulator-informed residual losses.

ECG-to-PPG Forward Mapper. We independently train an ECG-to-PPG conditional latent rectified flow G_ρ on paired ECG–PPG windows from the MC-MED training split, with the PAE frozen. Its flow-matching objective conditions on the ECG latent and targets the paired PPG latent. During PPG2ECG training, the decoded terminal ECG $\tilde{x}_e$ induces the pulse waveform used in the PPG-side simulator residual,

$$\tilde{x}_p = G_\rho(\tilde{x}_e). \tag{23}$$

We integrate the mapper's vector field with 10 Euler steps and decode the resulting PPG latent with the frozen PAE. The mapper's parameters remain fixed while the PPG-side residual backpropagates through G_ρ to the ECG generator. Inference from PPG omits G_ρ.

Simulator-guided Residual Losses. To capture heterogeneous electro-hemodynamic patterns, we fit simulator parameters using ICD-stratified training groups. For each group κ, we estimate $\eta^\kappa = (\eta^\kappa_e, \eta^\kappa_p)$ from representative ECG–PPG beats. These parameters are used only during training: they define the simulator residuals below, and their delay δ_{PAT} also provides the target of $\mathcal{L}_{\mathrm{PAT}}$ in PAE training. ICD labels are not used as model inputs, and PPG2ECG requires only PPG at inference time. Let $\mathcal{C}_e(\tilde{\mathbf{x}}^t_e) = \tilde{\mathbf{h}}_e \in \mathbb{R}^{L^c_e}$ denote a QRS-aligned generated ECG beat crop. Given fitted ECG simulator parameters η^κ_e and reference phase states $(x^\kappa_{e,\ell}, y^\kappa_{e,\ell})$, we penalize deviations from the ECG vector field by the Euler residual:

$$\mathcal{L}^e_{\mathrm{sim}} = \frac{1}{L^c_e - 1} \sum_{\ell=1}^{L^c_e - 1} \left\| \frac{\tilde{h}_{e,\ell+1} - \tilde{h}_{e,\ell}}{\Delta t_e} - f_e\left(x^\kappa_{e,\ell}, y^\kappa_{e,\ell}, \tilde{h}_{e,\ell}, t_{e,\ell}; \eta^\kappa_e\right) \right\|_2^2. \tag{24}$$

This term encourages generated ECG beats to follow physiologically plausible P–QRS–T dynamics rather than only matching point-wise waveform statistics.

Similarly, let $\mathcal{C}_p(\tilde{\mathbf{x}}^t_p) = \tilde{\mathbf{h}}_p \in \mathbb{R}^{L^c_p}$ denote the induced PPG beat crop. Given fitted PPG simulator parameters η^κ_p and reference phase states $(x^\kappa_{p,\ell}, y^\kappa_{p,\ell})$, we define

$$\mathcal{L}^p_{\mathrm{sim}} = \frac{1}{L^c_p - 1} \sum_{\ell=1}^{L^c_p - 1} \left\| \frac{\tilde{h}_{p,\ell+1} - \tilde{h}_{p,\ell}}{\Delta t_p} - f_p\left(x^\kappa_{p,\ell}, y^\kappa_{p,\ell}, \tilde{h}_{p,\ell}, t_{p,\ell}; \eta^\kappa_p\right) \right\|_2^2. \tag{25}$$

This term regularizes the generated ECG through its induced PPG response, encouraging compatibility with delayed peripheral pulse dynamics.

The complete simulator-guided flow objective is

$$\mathcal{L}_{\mathrm{Flow}} = \mathcal{L}_{\mathrm{RF}} + \lambda_e \mathcal{L}^e_{\mathrm{sim}} + \lambda_p \mathcal{L}^p_{\mathrm{sim}}. \tag{26}$$

The simulator-informed losses and the ECG-to-PPG forward mapper are used only during training. At inference time, the model only requires the PPG input, solves the learned latent flow, and decodes the generated ECG latent through the frozen ECG decoder.

3.4 Training and Inference

PPG2ECG is optimized in two phases. We first train the Physiology-Aware AutoEncoder with the objective $\mathcal{L}_{\mathrm{PAE}}$ to obtain aligned PPG and ECG latent representations. Afterward, we freeze the PPG encoder, ECG encoder, and ECG decoder, and train only the PPG-conditioned latent rectified flow. The simulator-informed terms are used only during training and do not change the inference procedure. At inference time, only the PPG input is required. Given $\mathbf{x}_p$, we compute the condition latent $\mathbf{z}_p = E^p_\phi(\mathbf{x}_p)$, sample $\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and solve the learned ODE $d\mathbf{z}_t/dt = \mathbf{v}_\psi(\mathbf{z}_t, t, \mathbf{z}_p)$ from $t = 0$ to $t = 1$. With an Euler solver using N steps and step size $\Delta t = 1/N$, the update is

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \Delta t\, \mathbf{v}_\psi(\mathbf{z}_k, t_k, \mathbf{z}_p), \qquad k = 0, \ldots, N-1. \tag{27}$$

The final latent $\mathbf{z}_N$ is decoded by the frozen ECG decoder to obtain the generated ECG waveform:

$$\hat{\mathbf{x}}_e = D^e_\theta(\mathbf{z}_N). \tag{28}$$

4 Experiments

Dataset and Preprocessing. We evaluate PPG2ECG on MC-MED [25], a large-scale multimodal clinical monitoring dataset comprising 118,385 adult emergency department visits from 70,545 unique patients at Stanford Health Care between 2020 and 2022. This dataset provides continuous, temporally synchronized physiological waveforms—specifically electrocardiography (ECG) and photoplethysmography (PPG)—alongside visit-level clinical metadata. MC-MED records PPG at 125 Hz and lead II ECG at 500 Hz [15]. We use preprocessed recordings resampled to 40 Hz for PPG and 120 Hz for ECG. Each synchronized 10-second window thus contains 400 PPG samples ($\mathbf{x}_p \in \mathbb{R}^{400\times 1}$) and 1200 ECG samples ($\mathbf{x}_e \in \mathbb{R}^{1200\times 1}$). Prior to modeling, we apply z-score normalization to each segment. Finally, to ensure signal integrity, we explicitly exclude any windows exhibiting missing values, severe truncation, or invalid signal lengths. We use the official MC-MED random patient-level split, which keeps all visits from each patient in the same partition.

Baselines. We compare four baselines: RDDM [47] models regions of interest (such as QRS complexes) separately from other regions with diffusion; CardioGAN [45] uses an attention-based generator with time- and frequency-domain discriminators; CardioFlow [38] applies rectified flow to waveforms; and PPGFlowECG [15] applies it in a shared PPG–ECG latent space. All models are scored on the same 10-second MC-MED windows at 120 Hz.

Ablation Study. To isolate the impact of our core physiological designs, we evaluate PPG2ECG against two ablated variants: (i) *PPG2ECG w/o Sim*: We remove all simulator guidance during latent flow training. This ablation measures the contribution of simulator guidance during latent flow training. (ii) *PPG2ECG w/o PAE*: We replace our modality-specific-to-shared PAE architecture with two separate standard autoencoders. This validates the importance of progressively mapping acquisition-specific structures into a shared cardiovascular latent space rather than processing modalities entirely independently.

4.1 ECG Generation Performance

Waveform-Level Fidelity. MAE and RMSE measure sample-level error; FD and FID compare ECG distributions in waveform space and in a pretrained encoder's feature space, respectively. On MC-MED, PPG2ECG reduces MAE from 0.87 to 0.71 relative to PPGFlowECG and also lowers FID and heart-rate MAE (in bpm; Table 1). Correlation between aligned samples remains modest (Pearson $r = 0.1906$), but R-peak F1 for beats matched within 50 ms is 0.74 for PPG2ECG versus 0.28 for CardioFlow. Without simulator guidance, F1 drops to 0.62.

Table 1: ECG generation on MC-MED (mean $\pm$ SD over three runs).

Method	MAE ↓	RMSE ↓	FD ↓	FID ↓	HR MAE ↓	Pearson r ↑	R-peak F1 ↑
RDDM	0.95 ± 0.01	1.41 ± 0.01	60.34 ± 0.47	47.97 ± 0.12	5.48 ± 0.08	0.0074 ± 0.0006	0.26 ± 0.01
CardioGAN	1.07 ± 0.01	1.46 ± 0.01	74.17 ± 0.22	47.01 ± 0.01	4.71 ± 0.01	-0.0040 ± 0.0007	0.06 ± 0.01
CardioFlow	0.94 ± 0.01	1.35 ± 0.01	58.78 ± 0.65	86.77 ± 0.22	6.83 ± 0.74	0.0277 ± 0.0004	0.28 ± 0.01
PPGFlowECG	0.87 ± 0.04	1.32 ± 0.04	46.64 ± 1.49	44.17 ± 0.11	7.08 ± 0.10	0.0051 ± 0.0004	0.15 ± 0.01
PPG2ECG	$\mathbf{0.71 \pm 0.04}$	$\mathbf{1.07 \pm 0.05}$	$\mathbf{39.68 \pm 1.96}$	$\mathbf{37.79 \pm 0.27}$	$\mathbf{3.94 \pm 0.76}$	$\mathbf{0.1906 \pm 0.0005}$	$\mathbf{0.74 \pm 0.01}$
w/o Sim	0.84 ± 0.01	1.28 ± 0.01	40.51 ± 1.20	45.90 ± 0.27	4.94 ± 0.57	0.1212 ± 0.0018	0.62 ± 0.01
w/o PAE	0.85 ± 0.05	1.26 ± 0.01	43.10 ± 0.88	44.07 ± 0.37	6.36 ± 0.07	0.1471 ± 0.0022	0.71 ± 0.01

External Dataset Validation. Table 2 compares PPG2ECG with RDDM, CardioGAN, CardioFlow, and PPGFlowECG on three external datasets. On VitalDB [29], PPG2ECG has the lowest waveform MAE (0.61) and RMSE (1.30), while CardioGAN has the lowest heart-rate MAE (9.17 bpm). On MIMIC-AFib [3], PPG2ECG and PPGFlowECG are nearly tied on MAE and RMSE, but PPG2ECG has the lowest heart-rate MAE (10.41 bpm). On BIDMC [41], PPG2ECG has the lowest RMSE (0.83) and heart-rate MAE (3.47 bpm), while PPGFlowECG has a slightly lower waveform MAE (0.49 vs. 0.50).

Morphology and Interval Fidelity. On MC-MED, we also assess morphology and intervals beyond global waveform similarity [59]. We use NeuroKit2 [36] to locate fiducial points and compare the seven measurements in Table 3 between generated and real ECGs (Appendix C.4). As shown in

Table 2: ECG generation on three external datasets (HR MAE in bpm).

Method	VitalDB			MIMIC-AFib			BIDMC		
	MAE ↓	RMSE ↓	HR MAE ↓	MAE ↓	RMSE ↓	HR MAE ↓	MAE ↓	RMSE ↓	HR MAE ↓
RDDM	0.77±0.01	1.41±0.01	10.86±0.08	1.00±0.01	1.39±0.01	11.97±0.19	0.97±0.01	1.32±0.01	3.48±0.03
CardioGAN	0.82±0.01	1.41±0.01	**9.17**±0.06	1.03±0.01	1.41±0.01	12.68±0.15	1.02±0.01	1.43±0.01	4.35±0.01
CardioFlow	0.62±0.01	1.35±0.01	16.40±0.06	0.54±0.01	0.87±0.01	14.54±0.12	0.52±0.01	0.85±0.01	6.83±0.11
PPGFlowECG	0.62±0.01	1.31±0.01	13.30±0.09	0.52±0.01	0.84±0.01	12.81±0.18	**0.49**±0.01	0.84±0.01	6.54±0.01
PPG2ECG	**0.61**±0.01	**1.30**±0.01	10.49±0.02	**0.52**±0.01	**0.84**±0.01	**10.41**±0.11	0.50±0.01	**0.83**±0.01	**3.47**±0.01

Table 3, PPG2ECG achieves the best performance across all metrics. Compared with PPGFlowECG, PPG2ECG reduces PR error from 68.9 to 59.9 ms, QRS error from 48.1 to 38.6 ms, QT error from 32.5 to 29.3 ms, and QTcF error from 35.9 to 32.2 ms. It also improves ST-J60 deviation and both P- and T-wave duration errors, indicating that the generated ECGs better preserve conduction, depolarization, and ST-segment morphology.

Table 3: Morphology and interval fidelity on MC-MED. We report MAE in milliseconds for intervals and durations, and in normalized amplitude units for ST-J60 deviation.

Method	PR ↓	QRS ↓	QT ↓	QTcF ↓	ST-J60 ↓	P dur. ↓	T dur. ↓
RDDM	66.3 ± 0.4	39.2 ± 1.1	40.7 ± 0.6	43.8 ± 0.6	0.22 ± 0.0	19.6 ± 0.2	22.3 ± 0.1
CardioGAN	65.2 ± 0.6	43.0 ± 0.4	56.3 ± 0.8	61.5 ± 0.9	0.36 ± 0.02	27.6 ± 0.3	24.7 ± 0.2
CardioFlow	75.8 ± 1.5	56.8 ± 0.6	73.9 ± 1.4	81.1 ± 1.3	0.20 ± 0.01	52.8 ± 0.1	38.9 ± 0.2
PPGFlowECG	68.9 ± 0.5	48.1 ± 0.3	32.5 ± 0.3	35.9 ± 0.3	0.16 ± 0.01	28.8 ± 0.2	37.7 ± 0.2
PPG2ECG	$\mathbf{59.9 \pm 0.9}$	$\mathbf{38.6 \pm 0.5}$	$\mathbf{29.3 \pm 0.6}$	$\mathbf{32.2 \pm 0.5}$	$\mathbf{0.14 \pm 0.02}$	$\mathbf{19.3 \pm 0.4}$	$\mathbf{21.3 \pm 0.4}$

4.2 Cardiovascular Disease Detection

To assess whether the generated ECGs preserve clinically useful information, we evaluate downstream multi-label cardiovascular disease detection [64, 63, 62, 22]. Following prior PPG-to-ECG benchmarks [15], we use visit-level ICD-based cardiovascular labels, including I48, I71, I70, I44, I25, and I50, covering rhythm, conduction, vascular, ischemic, and heart-failure-related conditions.

Table 4: Quantitative evaluation of multi-label cardiovascular disease classification on the MC-MED dataset. We compare a PPG-based baseline (PaPaGei [40]) and PPG2ECG. Higher AUROC indicates better performance.

Methods	Disease Categories (AUROC) ↑						Macro-AUROC ↑
	I48	I71	I70	I44	I25	I50	
PaPaGei	0.525 ± 0.03	0.521 ± 0.11	0.513 ± 0.03	0.653 ± 0.11	0.528 ± 0.07	0.520 ± 0.03	0.543 ± 0.04
PPG2ECG	$\mathbf{0.697 \pm 0.01}$	$\mathbf{0.610 \pm 0.02}$	$\mathbf{0.622 \pm 0.02}$	$\mathbf{0.656 \pm 0.03}$	$\mathbf{0.665 \pm 0.01}$	$\mathbf{0.551 \pm 0.02}$	$\mathbf{0.633 \pm 0.01}$

As shown in Table 4, ECGs generated by PPG2ECG achieve higher AUROC than the PPG-based PaPaGei [40] baseline across all disease categories, improving Macro-AUROC from 0.543 to 0.633. The gains are particularly clear for I48, I70, and I25, suggesting that the generated ECGs recover clinically relevant electrical patterns that are less accessible from raw PPG alone. We use the same data split and classifier architecture across all settings. Since the labels are visit-level ICD codes rather than ECG-only diagnostic annotations, this experiment is intended as a downstream clinical utility evaluation rather than evidence for standalone diagnostic deployment.

4.3 Qualitative Case Studies

Figure 2 shows a noisy reference ECG in Case 1 (107 bpm), broad QRS complexes in Case 2 (60 bpm), and a regular rhythm in Case 3 (99 bpm). PPG2ECG follows the reference beat timing and waveform shape in all three windows, with Pearson correlations of 0.84, 0.90, and 0.93, respectively. The four baselines generate recognizable beats, but their peaks often miss the reference peaks ($r \leq 0.21$ in these windows). We z-scored each signal within its window and plotted all ECGs on a common scale

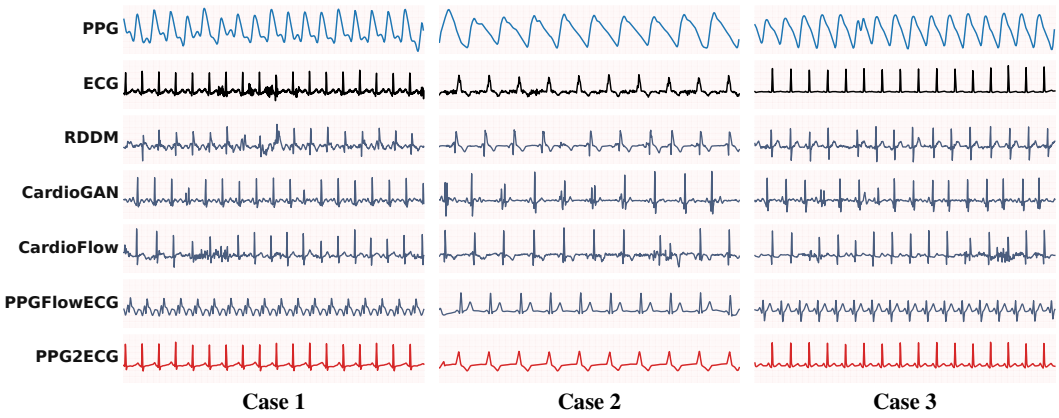

Figure 2: Input PPG, reference ECG, and ECGs generated by four baselines and PPG2ECG for three held-out 10-second MC-MED windows from different patients.

in the dataset's per-encounter QRS-upright orientation, without smoothing, time shifts, or further rescaling.

5 Conclusion

In this paper, we presented PPG2ECG, a physiology-guided latent rectified flow framework designed to overcome the limitations of purely data-driven PPG-to-ECG generation. Rather than treating this task as a simple statistical cross-modal translation, PPG2ECG explicitly grounds the generative process in the underlying cardiovascular dynamics. We introduced a novel electro-hemodynamic simulator that co-models ECG and PPG via shared cardiac phase dynamics. This simulator provides soft, phase-aware guidance to the Physiology-Aware AutoEncoder to structure the latent space, and explicitly regularizes the latent rectified flow to enforce both ECG-side morphological fidelity and forward ECG-to-PPG hemodynamic consistency. Extensive evaluations on the large-scale MC-MED dataset demonstrate that PPG2ECG consistently outperforms existing baselines, generating ECGs that are not only signal-faithful but also preserve critical clinical intervals and improve downstream cardiovascular disease classification. Ultimately, PPG2ECG effectively bridges the gap between ubiquitous wearable sensing and verifiable clinical cardiac assessment, demonstrating the profound value of integrating mechanistic physiological priors with modern latent generative modeling.

Acknowledgement

This research was partially supported by the U.S. National Science Foundation under Award Numbers 2616633, 2442172, 2312502, 2319449, 2211557, 2303037, 2312501, 2425919, 2413417, 2531008, and 2106859, and by the U.S. National Institutes of Health under Award Numbers K25DK135913, RF1NS139325, R01DK143456, U18DP006922, OT2OD038003, R01HL175135, U54OD036472, U54HG012517, and U24DK097771. This research was also partially supported by internal funds and GPU servers provided by the Computer Science Department of Emory University, the SRC JUMP 2.0 Center, Amazon Research Awards, Snapchat Gifts, Optum AI, NEC, the Easton Center, GPU resources from the iTiger GPU cluster [46], and the U.S. National Science Foundation ACCESS program through allocation CIS260707.

The authors declare no competing interests.

References

[1] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, 28(3):R1–R39, 2007.

[2] Xianglong Bao, Bosong Huang, Xiaoda Wang, Keqi Han, Carl Yang, Ming Jin, and Shirui Pan. Rethinking world models for biosignals. *arXiv preprint arXiv:2610.17510*, 2026.

[3] Syed Khairul Bashar, Eric Ding, Allan J. Walkey, David D. McManus, and Ki H. Chon. Noise detection in electrocardiogram signals for intensive care unit patients. *IEEE Access*, 7:88357–88368, 2019.

[4] Denisse Castaneda, Aibhlin Esparza, Mohammad Ghamari, Cinna Soltanpur, and Homer Nazeran. A review on wearable photoplethysmography sensors and their potential future applications in health care. *International Journal of Biosensors & Bioelectronics*, 4(4):195–202, 2018.

[5] Peter H. Charlton, Panicos A. Kyriacou, Jonathan Mant, Vaidotas Marozas, Phil Chowienczyk, and Jordi Alastruey. Wearable photoplethysmography for cardiovascular monitoring. *Proceedings of the IEEE*, 110(3):355–381, 2022.

[6] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[7] Zhaoliang Chen, Cheng Ding, Saurabh Kataria, Runze Yan, Minxiao Wang, Randall Lee, and Xiao Hu. GPT-PPG: a GPT-based foundation model for photoplethysmography signals. *Physiological Measurement*, 46(5):055004, 2025.

[8] Zhaoliang Chen, Saurabh Kataria, Minxiao Wang, and Xiao Hu. Bridging ECG and PPG: Latent-space prediction for robust physiological analysis. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 10693–10703, 2026.

[9] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10850–10869, 2023.

[10] Zhengyao Ding, Yujian Hu, Ziyu Li, Yiheng Mao, Haitao Li, Dongchen Zhou, Xuesen Chu, Long Yu, Ziyi Liu, Fei Wu, et al. AI modeling photoplethysmography to electrocardiography useful for predicting cardiovascular disease. *npj Digital Medicine*, 9(1):61, 2026.

[11] Jan-Christoph Edelmann, Dominik Mair, Daniel Ziesel, Martin Burtscher, and Thomas Ussmueller. An ECG simulator with a novel ECG profile for physiological signals. *Journal of Medical Engineering & Technology*, 42(7):501–509, 2018.

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, volume 235, pages 12606–12633, 2024.

[13] Xiaocheng Fang, Jieyi Cai, Guangkun Nie, Haoyu Wang, Jiarui Jin, Yujie Xiao, Bo Liu, Chenyang He, Qinghao Zhao, Gaofeng Cheng, et al. EchoBridge: Long-tail-aware ECG-echocardiography text alignment for echocardiography-derived cardiac findings. *arXiv preprint arXiv:2607.24553*, 2026.

[14] Xiaocheng Fang, Zhengyao Ding, Guangkun Nie, Jieyi Cai, Yujie Xiao, Bo Liu, Jiarui Jin, Haoyu Wang, Shun Huang, Ting Chen, et al. ECGFlowCMR: Pretraining with ECG-generated cine CMR helps cardiac disease classification and phenotype prediction. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 10877–10888, 2026.

[15] Xiaocheng Fang, Jiarui Jin, Haoyu Wang, Che Liu, Jieyi Cai, Yujie Xiao, Guangkun Nie, Bo Liu, Shun Huang, Hongyan Li, et al. PPGFlowECG: Latent rectified flow with cross-modal encoding for PPG-guided ECG generation and cardiovascular disease detection. *arXiv preprint arXiv:2509.19774*, 2025.

[16] Jesse Fine, Kimberly L. Branan, Andres J. Rodriguez, Tananant Boonya-ananta, Ajmal, Jessica C. Ramella-Roman, Michael J. McShane, and Gerard L. Coté. Sources of inaccuracy in photoplethysmography for continuous cardiovascular monitoring. *Biosensors*, 11(4):126, 2021.

[17] Tomer Golany, Daniel Freedman, and Kira Radinsky. ECG ODE-GAN: Learning ordinary differential equations of ECG dynamics via generative adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 134–141, 2021.

[18] Tomer Golany, Kira Radinsky, and Daniel Freedman. SimGANs: Simulator-based generative adversarial networks for ECG synthesis to improve deep ECG classification. In *International Conference on Machine Learning*, volume 119, pages 3597–3606, 2020.

[19] Chenyang He, Xinyi Shao, Shun Huang, Bosong Huang, Daoqiang Zhang, Ming Jin, and Cheng Ding. CAP: Towards PPG universal representation learning with patient-level supervision. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 11050–11060, 2026.

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

[22] Bosong Huang, Panzhen Zhao, Zengxiang Li, Patricia Lee, Wei Jin, Alan Wee-Chung Liew, Ming Jin, and Shirui Pan. LVCG: Learning ECG representations in the latent vectorcardiogram space. In *International Conference on Machine Learning*, volume 306, pages 48484–48505, 2026.

[23] Jiarui Jin, Haoyu Wang, Hongyan Li, Jun Li, Jiahui Pan, and Shenda Hong. Reading your heart: Learning ECG words and sentences via pre-training ECG language model. In *International Conference on Learning Representations*, 2025.

[24] Jiarui Jin, Haoyu Wang, Xingliang Wu, Xiaocheng Fang, Xiang Lan, Zihan Wang, Deyun Zhang, Bo Liu, Yingying Zhang, Xian Wu, et al. ECG-R1: Protocol-guided and modality-agnostic MLLM for reliable ECG interpretation. In *International Conference on Machine Learning*, volume 306, pages 54089–54130, 2026.

[25] Aman Kansal, Emma Chen, Boyang Tom Jin, Pranav Rajpurkar, and David A. Kim. MC-MED, multimodal clinical monitoring in the emergency department. *Scientific Data*, 12(1):1094, 2025.

[26] Selcan Kaplan Berkaya, Alper Kursat Uysal, Efnan Sora Gunal, Semih Ergin, Serkan Gunal, and M. Bilginer Gulmezoglu. A survey on ECG analysis. *Biomedical Signal Processing and Control*, 43:216–235, 2018.

[27] Paul Kligfield, Leonard S. Gettes, James J. Bailey, Rory Childers, Barbara J. Deal, E. William Hancock, Gerard van Herpen, Jan A. Kors, Peter Macfarlane, David M. Mirvis, et al. Recommendations for the standardization and interpretation of the electrocardiogram: Part I: The electrocardiogram and its technology. *Circulation*, 115(10):1306–1324, 2007.

[28] Ella Lan. Performer: A novel PPG-to-ECG reconstruction transformer for a digital biomarker of cardiovascular disease detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1991–1999, 2023.

[29] Hyung-Chul Lee, Yoonsang Park, Soo Bin Yoon, Seong Mi Yang, Dongnyeok Park, and Chul-Woo Jung. VitalDB, a high-fidelity multi-parameter vital signs database in surgical patients. *Scientific Data*, 9:279, 2022.

[30] Sangyun Lee, Zinan Lin, and Giulia Fanti. Improving the training of rectified flows. In *Advances in Neural Information Processing Systems*, volume 37, pages 63082–63109, 2024.

[31] Xiaoyan Li, Shixin Xu, Faisal Habib, Neda Aminnejad, Arvind Gupta, and Huaxiong Huang. CLEP-GAN: an innovative approach to subject-independent ECG reconstruction from PPG signals. *BMC Bioinformatics*, 26(1):306, 2025.

[32] Zhiyuan Li, Yuanyuan Tian, Yanrui Jin, Xiaoyang Wei, Mengxiao Wang, Jinlei Liu, and Chengliang Liu. EDDM: A novel ECG denoising method using dual-path diffusion model. *IEEE Transactions on Instrumentation and Measurement*, 74:1–15, 2025.

[33] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

[34] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

[35] Zewen Liu, Xiaoda Wang, Bohan Wang, Zijie Huang, Carl Yang, and Wei Jin. Graph ODEs and beyond: A comprehensive survey on integrating differential equations with graph neural networks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 6118–6128, 2025.

[36] Dominique Makowski, Tam Pham, Zen J. Lau, Jan C. Brammer, François Lespinasse, Hung Pham, Christopher Schölzel, and S. H. Annabel Chen. NeuroKit2: A Python toolbox for neurophysiological signal processing. *Behavior Research Methods*, 53(4):1689–1696, 2021.

[37] Patrick E. McSharry, Gari D. Clifford, Lionel Tarassenko, and Leonard A. Smith. A dynamical model for generating synthetic electrocardiogram signals. *IEEE Transactions on Biomedical Engineering*, 50(3):289–294, 2003.

[38] Yuta Nambu, Masahiro Kohjima, and Ryuji Yamamoto. CardioFlow: Learning to generate ECG from PPG with rectified flow. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.

[39] Guangkun Nie, Xiaocheng Fang, Gongzheng Tang, Yujie Xiao, Jun Li, Bo Liu, Hongyan Li, and Shenda Hong. AnyPPG: An ECG-guided PPG foundation model trained on over 100,000 hours of recordings for holistic health profiling. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 11717–11727, 2026.

[40] Arvind Pillai, Dimitris Spathis, Fahim Kawsar, and Mohammad Malekzadeh. PaPaGei: Open foundation models for optical physiological signals. In *International Conference on Learning Representations*, 2025.

[41] Marco A. F. Pimentel, Alistair E. W. Johnson, Peter H. Charlton, Drew Birrenkott, Peter J. Watkinson, Lionel Tarassenko, and David A. Clifton. Toward a robust estimation of respiratory rate from pulse oximeters. *IEEE Transactions on Biomedical Engineering*, 64:1914–1923, 2017.

[42] Wei Qian, Gaoji Su, Dan Guo, Jinxing Zhou, Xiaobai Li, Bin Hu, Shengeng Tang, and Meng Wang. PhysDiff: Physiology-based dynamicity disentangled diffusion model for remote physiological measurement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 6568–6576, 2025.

[43] Christopher Rackauckas, Yingbo Ma, Julius Martensen, Collin Warner, Kirill Zubov, Rohit Supekar, Dominic Skinner, Ali Ramadhan, and Alan Edelman. Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*, 2020.

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[45] Pritam Sarkar and Ali Etemad. CardioGAN: Attentive generative adversarial network with dual discriminators for synthesis of ECG from PPG. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 488–496, 2021.

[46] Mayira Sharif, Guangzeng Han, Weisi Liu, and Xiaolei Huang. Cultivating multidisciplinary AI workforce development on iTiger GPU cluster: Practices and challenges. In *Proceedings of the Practice and Experience in Advanced Research Computing 2026: Resilient Roots + Empowered Communities*, pages 1–6, 2026.

[47] Debaditya Shome, Pritam Sarkar, and Ali Etemad. Region-disentangled diffusion model for high-fidelity PPG-to-ECG translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15009–15019, 2024.

[48] Zitao Shuai, Zongzhe Xu, Yuntian Wu, Sirui Li, Tianhong Li, and Yuzhe Yang. Signal or noise? Understanding generative models for real-world sensor time series. *arXiv preprint arXiv:2607.04245*, 2026.

[49] Zitao Shuai, Zongzhe Xu, David Yang, Wei Wang, and Yuzhe Yang. OSF: On pre-training and scaling of sleep foundation models. In *International Conference on Machine Learning*, volume 306, pages 112847–112871, 2026.

[50] George Stoica, Vivek Ramanujan, Xiang Fan, Ali Farhadi, Ranjay Krishna, and Judy Hoffman. Contrastive flow matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1185–1194, 2025.

[51] Qunfeng Tang, Zhencheng Chen, Rabab Ward, and Mohamed Elgendi. Synthetic photoplethysmogram generation using two Gaussian functions. *Scientific Reports*, 10(1):13883, 2020.

[52] Qunfeng Tang, Zhencheng Chen, Rabab Ward, Carlo Menon, and Mohamed Elgendi. PPG2ECGps: An end-to-end subject-specific deep neural network model for electrocardiogram reconstruction from photoplethysmography signals without pulse arrival time adjustments. *Bioengineering*, 10(6):630, 2023.

[53] Minxiao Wang, Saurabh Kataria, Juntong Ni, Timothy G. Buchman, Jocelyn Grunwell, Mark Mai, Wei Jin, Matthew Clark, Stephanie Brown, Michael Fundora, et al. A unified AI approach for continuous monitoring of human health and diseases from intensive care unit to home with physiological foundation models (UNIPHY+). *arXiv preprint arXiv:2509.16348*, 2025.

[54] Xiaoda Wang, Ching Chang, Defu Cao, Kaiqiao Han, Fang Sun, Yue Huang, Minxiao Wang, Chang Xu, Xiao Luo, Runze Yan, et al. Beyond prediction: Toward verifiable physiological waveform reasoning: Advances, challenges, and future directions. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 13375–13378, 2026.

[55] Xiaoda Wang, Ching Chang, Defu Cao, Kaiqiao Han, Fang Sun, Yue Huang, Minxiao Wang, Chang Xu, Xiao Luo, Runze Yan, et al. Position: Beyond prediction: Toward verifiable physiological waveform reasoning with foundation models and agentic LLMs. In *International Conference on Machine Learning*, volume 306, pages 172441–172459, 2026.

[56] Xiaoda Wang, Kaiqiao Han, Yuhao Xu, Xiao Luo, Yizhou Sun, Wei Wang, and Carl Yang. SE-Diff: Simulator and experience enhanced diffusion model for comprehensive ECG generation. In *International Conference on Learning Representations*, 2026.

[57] Xiaoda Wang, Minxiao Wang, Maxwell A. Xu, Patrick Langer, Kaiqiao Han, Defu Cao, Xiao Luo, Yuzhe Yang, Yan Liu, Xiao Hu, et al. PPG-LM: A photoplethysmography-language model with multi-level clinical alignment. *arXiv preprint arXiv:2609.33516*, 2026.

[58] Xiaoda Wang, Yuji Zhao, Kaiqiao Han, Xiao Luo, Sanne van Rooij, Jennifer Stevens, Lifang He, Liang Zhan, Yizhou Sun, Wei Wang, et al. Conditional neural ODE for longitudinal Parkinson's disease progression forecasting. In *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 4131–4134, 2025.

[59] Muyan Weng, Defu Cao, Wei Yang, Yashaswi Sharma, and Yan Liu. TemporalBench: A benchmark for evaluating LLM-based agents on contextual and event-informed time series tasks. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 9997–10008, 2026.

[60] Jonas Wildberger, Maximilian Dax, Simon Buchholz, Stephen Green, Jakob H. Macke, and Bernhard Schölkopf. Flow matching for scalable simulation-based inference. In *Advances in Neural Information Processing Systems*, volume 36, pages 16837–16864, 2023.

[61] Guanchen Wu, Zhe Huang, Yuzhang Xie, Runze Yan, Akul Chopra, Deqiang Qiu, Xiao Hu, Fei Wang, and Carl Yang. MIRAGE: Knowledge graph-guided cross-cohort MRI synthesis for Alzheimer's disease prediction. *arXiv preprint arXiv:2603.02434*, 2026.

[62] Yuzhang Xie, Yuhua Wu, Ruiyu Wang, Fadi Nahab, Xiao Hu, and Carl Yang. Enhanced atrial fibrillation prediction in ESUS patients with hypergraph-based pre-training. *AMIA Joint Summits on Translational Science Proceedings*, 2026:526–535, 2026.

[63] Yuhao Xu, Xiaoda Wang, Jiaying Lu, Sirui Ding, Defu Cao, Huaxiu Yao, Yan Liu, Xiao Hu, and Carl Yang. EnECG: Efficient ensemble learning for electrocardiogram multi-task foundation model. *arXiv preprint arXiv:2511.22935*, 2025.

[64] Yuhao Xu, Xiaoda Wang, Yi Wu, Wei Jin, Xiao Hu, and Carl Yang. ECG-MoE: Mixture-of-expert electrocardiogram foundation model. *arXiv preprint arXiv:2603.04589*, 2026.

[65] Zongzhe Xu, Aakarsh Anand, Sarah Jiang, Chuntung Zhuang, Zitao Shuai, Sriram Sankararaman, and Yuzhe Yang. Inertia-1: An open exploration of wearable motion foundation models. In *Advances in Neural Information Processing Systems*, 2026.

[66] Zongzhe Xu, Zitao Shuai, Eideen Mozaffari, Ravi Shankar Aysola, Rajesh Kumar, and Yuzhe Yang. SleepLM: Natural-language intelligence for human sleep. In *International Conference on Machine Learning*, volume 306, pages 141707–141738, 2026.

[67] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

[68] Lige Zhang, Ziwei Huang, Xiaoda Wang, Gelei Xu, Xiao Hu, Carl Yang, and Wei Jin. Deep learning for health time series: Physiological signals, clinical records, and epidemic dynamics. *Preprints.org preprint, doi:10.20944/preprints202610.0067.v1*, 2026.

A The Electro-Hemodynamic Physiological Simulator

A.1 Euler Discretization

We solve the electro-hemodynamic simulator with a fixed-step explicit Euler method. For any state variable $u(t)$ satisfying $du/dt = v(t)$, the finite-difference approximation is

$$\frac{du}{dt}(t) \approx \frac{u(t+\Delta t) - u(t)}{\Delta t}, \tag{29}$$

which gives the Euler update

$$u(t+\Delta t) = u(t) + v(t)\Delta t. \tag{30}$$

For the coupled simulator state $(x(t), y(t), e(t), p(t))$, at time $t_\ell = \ell\Delta t$, the vector field is

$$\mathbf{v}_\ell = (f_x(x_\ell, y_\ell), f_y(x_\ell, y_\ell), f_e(x_\ell, y_\ell, e_\ell, t_\ell; \eta_e), f_p(x_\ell, y_\ell, p_\ell, t_\ell; \eta_p)) \,. \tag{31}$$

The Euler updates are

$$x_{\ell+1} = x_\ell + f_x(x_\ell, y_\ell)\Delta t, \tag{32}$$
$$y_{\ell+1} = y_\ell + f_y(x_\ell, y_\ell)\Delta t, \tag{33}$$
$$e_{\ell+1} = e_\ell + f_e(x_\ell, y_\ell, e_\ell, t_\ell; \eta_e)\Delta t, \tag{34}$$
$$p_{\ell+1} = p_\ell + f_p(x_\ell, y_\ell, p_\ell, t_\ell; \eta_p)\Delta t. \tag{35}$$

A.2 Fitting Objective

Motivation. The electro-hemodynamic simulator contains coupled electrical and hemodynamic components: the ECG readout represents upstream cardiac electrical activation, while the PPG readout represents the delayed and smoothed peripheral vascular response. Directly fitting this simulator with a single point-wise waveform loss is often insufficient. A waveform loss may match the global amplitude envelope while missing sharp QRS transitions, systolic upstrokes, or physiologically meaningful event timing. We therefore optimize the simulator parameters with a composite objective that combines waveform-level reconstruction, derivative consistency, and event-level peak alignment. This design encourages the fitted parameters to reproduce both the electrical source dynamics and the downstream hemodynamic response.

Let $e(t)$ and $p(t)$ denote the reference ECG and PPG waveforms, and let $\hat{e}_\eta(t)$ and $\hat{p}_\eta(t)$ denote the simulated ECG and PPG waveforms generated by the electro-hemodynamic simulator with parameters $\eta = (\eta_e, \eta_p)$. The total fitting objective is

$$\mathcal{L}_{\text{fit}}(\eta) = \lambda_{\text{ecg}}\mathcal{L}_{\text{ecg}} + \lambda_{\text{ppg}}\mathcal{L}_{\text{ppg}} + \lambda_{\text{deriv}}\mathcal{L}_{\text{deriv}} + \lambda_{\text{peak}}\mathcal{L}_{\text{peak}}. \tag{36}$$

Unless otherwise specified, we use $\lambda_{\text{ecg}} = 5.0$, $\lambda_{\text{ppg}} = 0.25$, $\lambda_{\text{deriv}} = 3.0$, and $\lambda_{\text{peak}} = 12.0$.

ECG waveform reconstruction. The ECG reconstruction term anchors the simulator to the electrical component of the cardiac cycle:

$$\mathcal{L}_{\text{ecg}} = \frac{1}{T_e}\sum_{t=1}^{T_e} \|\hat{e}_\eta(t) - e(t)\|_2^2 \,. \tag{37}$$

We assign this term a relatively large weight because the ECG is the primary physiological source that drives the downstream hemodynamic response. In practice, this term stabilizes the fitting of QRS-dominated electrical morphology and prevents the simulator from matching peripheral PPG morphology at the expense of unrealistic ECG dynamics.

PPG waveform reconstruction. The PPG reconstruction term measures the discrepancy between the simulated and reference peripheral pulse waveforms:

$$\mathcal{L}_{\text{ppg}} = \frac{1}{T_p}\sum_{t=1}^{T_p} \|\hat{p}_\eta(t) - p(t)\|_2^2 \,. \tag{38}$$

Compared with ECG, PPG is an indirect observation affected by pulse transit time, vascular compliance, peripheral filtering, sensor contact, and amplitude variation. Therefore, we use a smaller weight for this term. This allows PPG to guide the hemodynamic readout without dominating the optimization or distorting the fitted electrical source.

Derivative consistency. To preserve local dynamic structure, we further match temporal derivatives:

$$\mathcal{L}_{\text{deriv}} = \frac{1}{T_e - 1} \sum_{t=1}^{T_e-1} \|\nabla \hat{e}_\eta(t) - \nabla e(t)\|_2^2 + \frac{1}{T_p - 1} \sum_{t=1}^{T_p-1} \|\nabla \hat{p}_\eta(t) - \nabla p(t)\|_2^2 , \tag{39}$$

where $\nabla x(t) = x(t+1) - x(t)$ denotes the first-order finite difference. This term emphasizes fast physiological transitions, such as steep QRS slopes in ECG and the systolic upstroke in PPG. It is also less sensitive to slow baseline drift and global amplitude offsets than pure waveform reconstruction. The derivative loss therefore encourages the simulator to reproduce local transition speed and morphology, rather than only matching low-frequency signal shape.

Peak alignment. Event timing is central to electro-hemodynamic coupling. We therefore include a peak alignment term that focuses on local neighborhoods around detected physiological events, such as ECG R-peaks and PPG systolic peaks. Let $\mathcal{R}$ denote the set of detected ECG R-peak indices and $\mathcal{S}$ denote the set of detected PPG systolic peak indices. For an event index τ, let $\Omega(\tau)$ denote a local temporal window around the event. In our implementation, the window is defined by 0.20 seconds before and 0.60 seconds after the event. The peak alignment loss is

$$\mathcal{L}_{\text{peak}} = \frac{1}{|\mathcal{R}|} \sum_{\tau \in \mathcal{R}} \frac{1}{|\Omega_e(\tau)|} \sum_{t \in \Omega_e(\tau)} \|\hat{e}_\eta(t) - e(t)\|_2^2 + \frac{1}{|\mathcal{S}|} \sum_{\tau \in \mathcal{S}} \frac{1}{|\Omega_p(\tau)|} \sum_{t \in \Omega_p(\tau)} \|\hat{p}_\eta(t) - p(t)\|_2^2 . \tag{40}$$

This term receives the largest weight because peak locations define key physiological timing events. Matching peaks only through global waveform losses can be unstable: a small temporal shift may cause a large point-wise error even when the morphology is otherwise similar. By explicitly emphasizing local event neighborhoods, the peak loss encourages beat-level synchronization and improves the fitted pulse-arrival relationship between ECG and PPG.

Staged electro-hemodynamic optimization. We use a simple curriculum to stabilize fitting. During the initial portion of training, controlled by $\rho_{\text{ecg}} = 0.5$, we optimize only ECG-related terms:

$$\mathcal{L}_{\text{warmup}}(\eta) = \lambda_{\text{ecg}} \mathcal{L}_{\text{ecg}} + \lambda_{\text{deriv}} \mathcal{L}^e_{\text{deriv}} + \lambda_{\text{peak}} \mathcal{L}^e_{\text{peak}}, \tag{41}$$

where $\mathcal{L}^e_{\text{deriv}}$ and $\mathcal{L}^e_{\text{peak}}$ denote the ECG components of the derivative and peak losses. After this warm-up stage, we optimize the full electro-hemodynamic objective in Eq. 36. This staged design first stabilizes the cardiac electrical source and then introduces the more indirect PPG constraints. It reduces poor local minima in which the simulator prematurely fits delayed peripheral pulse morphology while the underlying ECG dynamics remain inaccurate.

The composite fitting objective assigns different roles to complementary physiological constraints. The ECG waveform loss stabilizes the electrical source, the PPG waveform loss guides the delayed vascular response, the derivative loss preserves fast local dynamics, and the peak alignment loss enforces event-level timing. Together, these terms produce simulator parameters that better capture both ECG morphology and ECG-to-PPG hemodynamic coupling, which are later used as physiological anchors for simulator-guided latent flow training.

ICD-stratified simulator fitting. We use the top-100 most frequent ICD-10 codes in the training split to define simulator fitting groups. For each group, representative ECG–PPG beat crops are collected from training samples and used to fit the group-specific simulator parameters $\eta^\kappa = (\eta^\kappa_e, \eta^\kappa_p)$ with the composite objective in Eq. 36, including the staged warm-up in Eq. 41. The fitted parameters are then frozen and used only during training: the fitted pulse-arrival delay δ_{PAT} serves as the target of the phase-delay consistency loss $\mathcal{L}_{\text{PAT}}$ in PAE training, and η^κ defines the simulator residual losses $\mathcal{L}^e_{\text{sim}}$ and $\mathcal{L}^p_{\text{sim}}$ in latent flow training. ICD labels are never used as inputs to the generator and are not required during inference.

A.3 Consistency Interpretation of Euler Residual Losses

The fitted electro-hemodynamic parameters are used to define simulator vector-field residuals on terminal generated beats during latent-flow training. We provide a simple consistency interpretation showing that minimizing these residuals keeps the generated beat close to a trajectory induced by the physiological simulator. For modality $m \in \{e, p\}$, let $\tilde{h}^m = (\tilde{h}^m_1, \ldots, \tilde{h}^m_{L_m})$ denote a generated beat

crop, where $m = e$ corresponds to ECG and $m = p$ corresponds to PPG. Let f_m be the corresponding simulator vector field with fitted parameters η_m, and let $s_\ell^m = (x_\ell^m, y_\ell^m)$ denote the reference cardiac phase state at step ℓ. The Euler residual is

$$r_\ell^m = \frac{\tilde{h}_{\ell+1}^m - \tilde{h}_\ell^m}{\Delta t_m} - f_m(s_\ell^m, \tilde{h}_\ell^m, t_\ell^m; \eta_m). \tag{42}$$

Assume that f_m is K_m-Lipschitz in the waveform state and that the generated beat has bounded average Euler residual,

$$\frac{1}{L_m - 1} \sum_{\ell=1}^{L_m-1} \|r_\ell^m\|_2^2 \le \epsilon_m^2. \tag{43}$$

Let $h^{m,\star}$ be the simulator trajectory initialized at the same first sample,

$$h_1^{m,\star} = \tilde{h}_1^m, \qquad h_{\ell+1}^{m,\star} = h_\ell^{m,\star} + \Delta t_m f_m(s_\ell^m, h_\ell^{m,\star}, t_\ell^m; \eta_m). \tag{44}$$

Then the deviation between the generated beat and the simulator-induced trajectory satisfies

$$\max_{\ell \le L_m} \|\tilde{h}_\ell^m - h_\ell^{m,\star}\|_2 \le \Delta t_m \exp(K_m T_m) \sum_{\ell=1}^{L_m-1} \|r_\ell^m\|_2, \tag{45}$$

where $T_m = (L_m - 1)\Delta t_m$ is the beat duration. Consequently, by the Cauchy–Schwarz inequality and Eq. 43,

$$\sum_{\ell=1}^{L_m-1} \|r_\ell^m\|_2 \le \sqrt{L_m - 1} \left(\sum_{\ell=1}^{L_m-1} \|r_\ell^m\|_2^2 \right)^{1/2} \le (L_m - 1)\, \epsilon_m, \tag{46}$$

so that

$$\max_{\ell \le L_m} \|\tilde{h}_\ell^m - h_\ell^{m,\star}\|_2 \le T_m \exp(K_m T_m) \epsilon_m. \tag{47}$$

Proof. Let $\delta_\ell = \|\tilde{h}_\ell^m - h_\ell^{m,\star}\|_2$. By the definition of r_ℓ^m, the generated beat satisfies $\tilde{h}_{\ell+1}^m = \tilde{h}_\ell^m + \Delta t_m f_m(s_\ell^m, \tilde{h}_\ell^m, t_\ell^m; \eta_m) + \Delta t_m r_\ell^m$. Subtracting the Euler update of $h^{m,\star}$ and applying the triangle inequality,

$$\delta_{\ell+1} \le \delta_\ell + \Delta t_m \left\| f_m(s_\ell^m, \tilde{h}_\ell^m, t_\ell^m; \eta_m) - f_m(s_\ell^m, h_\ell^{m,\star}, t_\ell^m; \eta_m) \right\|_2 + \Delta t_m \|r_\ell^m\|_2. \tag{48}$$

Using the Lipschitz condition gives

$$\delta_{\ell+1} \le (1 + K_m \Delta t_m)\delta_\ell + \Delta t_m \|r_\ell^m\|_2. \tag{49}$$

Since $\delta_1 = 0$, unrolling this recursion yields, for every $\ell \le L_m$,

$$\delta_\ell \le \sum_{j=1}^{\ell-1} (1 + K_m \Delta t_m)^{\ell-1-j}\, \Delta t_m \|r_j^m\|_2 \le (1 + K_m \Delta t_m)^{L_m-2}\, \Delta t_m \sum_{j=1}^{L_m-1} \|r_j^m\|_2 \le \exp(K_m T_m)\, \Delta t_m \sum_{j=1}^{L_m-1} \|r_j^m\|_2, \tag{50}$$

where the last step uses $1 + u \le e^u$ and $(L_m - 2)\Delta t_m \le T_m$. Taking the maximum over ℓ gives Eq. 45, and combining it with Eq. 46 gives Eq. 47.

This result shows that the simulator-guided Euler residual losses do more than penalize point-wise waveform error. Under mild smoothness assumptions, small residuals imply that the generated ECG or PPG beat remains close to a trajectory generated by the fitted physiological vector field. Therefore, the ECG-side and PPG-side simulator losses constrain generated samples toward electrohemodynamically plausible dynamics.

B Why Physiology-Guided Latent Rectified Flow?

This section provides a concise theoretical justification for employing latent rectified flow within the PPG2ECG framework. Rather than asserting that rectified flow is universally superior to alternative generative models, our objective is to clarify why it serves as a principled and stable mechanism for PPG-conditioned ECG generation when operating within a structured physiological latent space.

Let $z_p = E_\phi^p(x_p)$ denote the PPG condition latent and $z_e = E_\phi^e(x_e)$ denote the paired ECG latent. Following the optimization of the Physiology-Aware AutoEncoder (PAE), PPG2ECG learns a conditional vector field that transports a Gaussian source latent $z_0 \sim \mathcal{N}(0, I)$ toward the ECG latent distribution, conditioned on z_p. For a time variable $t \sim \mathcal{U}(0, 1)$, the linear interpolation path is defined as

$$z_t = (1 - t)z_0 + tz_e,$$

with the target velocity $u_t = z_e - z_0$. The latent rectified-flow objective is then formulated as

$$\mathcal{L}_{\mathrm{RF}} = \mathbb{E}\left[\|v_\psi(z_t, t, z_p) - (z_e - z_0)\|_2^2\right].$$

Conditional transport interpretation. Given a fixed condition z_p, the population minimizer of $\mathcal{L}_{\mathrm{RF}}$ is

$$v^\star(z, t, z_p) = \mathbb{E}\left[z_e - z_0 \mid z_t = z, t, z_p\right].$$

This follows from the standard result that the conditional mean minimizes the squared-error regression objective, as in rectified flow [34] and flow matching [33]. Let $p_t(z \mid z_p)$ denote the marginal distribution of z_t conditioned on z_p. For any smooth test function φ, differentiating along the interpolation path yields

$$\frac{d}{dt}\mathbb{E}\left[\varphi(z_t) \mid z_p\right] = \mathbb{E}\left[\nabla_z \varphi(z_t)^\top (z_e - z_0) \mid z_p\right].$$

Substituting the optimal vector field $v^\star$ results in

$$\frac{d}{dt}\mathbb{E}\left[\varphi(z_t) \mid z_p\right] = \mathbb{E}\left[\nabla_z \varphi(z_t)^\top v^\star(z_t, t, z_p) \mid z_p\right],$$

which corresponds to the weak form of the continuity equation:

$$\partial_t p_t(z \mid z_p) + \nabla_z \cdot (p_t(z \mid z_p) v^\star(z, t, z_p)) = 0.$$

Consequently, under standard regularity assumptions on the vector field, simulating the ordinary differential equation (ODE) $\frac{dz_t}{dt} = v^\star(z_t, t, z_p)$ deterministically transports the Gaussian source distribution to the conditional ECG latent distribution associated with the observed PPG latent z_p. This marginal-preservation property holds for the population minimizer of $\mathcal{L}_{\mathrm{RF}}$ alone. The simulator terms introduced below deliberately bias the learned velocity field toward physiologically consistent terminal states, trading exact marginal matching for physiological plausibility.

Latent-space stability. The above derivation formally justifies the generative process in the latent space. To connect this latent transport to target waveform generation, we assume the frozen ECG decoder D_θ^e is L_D-Lipschitz continuous:

$$\|D_\theta^e(z_1) - D_\theta^e(z_2)\|_2 \le L_D \|z_1 - z_2\|_2.$$

Let $\hat{z}_e$ be the terminal latent produced by the learned flow and let $\hat{x}_e = D_\theta^e(\hat{z}_e)$. The waveform generation error can be bounded as

$$\|\hat{x}_e - x_e\|_2 \le L_D \|\hat{z}_e - z_e\|_2 + \|D_\theta^e(z_e) - x_e\|_2.$$

This inequality shows that the waveform error is upper-bounded by two components: the latent discrepancy $\|\hat{z}_e - z_e\|_2$ and the reconstruction error of the frozen autoencoder. Because PPG-to-ECG generation is a many-to-one inverse problem, the latent discrepancy contains an irreducible part: even a perfect flow samples $\hat{z}_e$ from the conditional distribution $p(z_e \mid z_p)$ rather than recovering the particular paired latent z_e, so this term reflects the conditional spread of plausible ECG latents in addition to any transport error. The bound nevertheless motivates the two-stage design: a PAE with small reconstruction error $\|D_\theta^e(z_e) - x_e\|_2$ keeps the second term small, so that the remaining waveform error is governed by the latent-space transport on which the rectified flow and the simulator guidance act.

Role of physiological guidance. The simulator-guided objective further regularizes this conditional transport:

$$\mathcal{L}_{\mathrm{Flow}} = \mathcal{L}_{\mathrm{RF}} + \lambda_e \mathcal{L}^e_{\mathrm{sim}} + \lambda_p \mathcal{L}^p_{\mathrm{sim}}.$$

In this formulation, $\mathcal{L}_{\mathrm{RF}}$ governs the statistical transport from the Gaussian prior to the empirical ECG latent distribution. Concurrently, the terms $\mathcal{L}^e_{\mathrm{sim}}$ and $\mathcal{L}^p_{\mathrm{sim}}$ bias the predicted terminal state toward trajectories that exhibit valid ECG morphology and forward ECG-to-PPG hemodynamic compatibility. Crucially, because these simulator constraints are applied exclusively during training via gradient signals along the flow trajectory, they do not strictly bound inference to a rigid simulator template. Instead, they function as soft physiological priors, narrowing the set of plausible solutions in the highly underdetermined PPG-to-ECG inverse problem.

In summary, PPG2ECG operates as a regularized conditional transport framework: the PAE constructs a dense, physiology-aware manifold, the rectified flow learns a marginal-preserving deterministic transport upon it, and the simulator guidance anchors the generated distributions to electro-hemodynamically plausible dynamics.

C Implementation Details

C.1 Training Setup

We use the official MC-MED random patient-level split. We fit the simulator and train all components of PPG2ECG exclusively on visits listed in `split_random_train.csv`. All MC-MED results in the main text and Appendix D use windows from visits listed in `split_random_test.csv`. We train in PyTorch on a single NVIDIA H200 GPU with batch size 4. We use PPG resampled to 40 Hz and ECG resampled to 120 Hz. We z-score each modality within each window. After fitting the simulator (Appendix A), we train the Physiology-Aware AutoEncoder (PAE). With the PAE frozen, we train the ECG-to-PPG conditional latent rectified flow G_ρ separately on paired training windows. We then train the PPG-to-ECG generator with both the PAE and G_ρ frozen. We apply $\mathcal{L}_{\mathrm{PAT}}$ only to training windows with heart rates of 50–100 bpm, retain windows outside this range for the primary rectified-flow objective, and generate ECGs at inference without estimating PAT.

Baseline evaluation. For RDDM, CardioGAN, CardioFlow, and PPGFlowECG, we use the authors' official implementations and released checkpoints with their recommended input preprocessing. We resample inputs to each model's required rate and evaluate a single generated ECG per input on the same 10-second windows at 120 Hz. All step-based methods use 10 sampling steps in the main comparison. For the external evaluation in Table 2, we use PPG2ECG and all four baselines without target-domain fine-tuning or calibration. The released CardioGAN weights were trained in part on BIDMC [45], so its BIDMC result includes prior exposure to that dataset.

C.2 Physiology-Aware AutoEncoder

The PAE maps the resampled PPG and ECG segments into a shared electro-hemodynamic latent space. The encoder first processes PPG and ECG with modality-specific stems, then passes the representations through shared blocks with lightweight modality-aware adapters, and finally maps them to a common Gaussian latent posterior. The decoder reconstructs waveforms at their respective input sampling rates using a residual reconstruction branch and a lightweight decomposition branch that models level, trend, and seasonal components. The phase estimators $\Theta_e(\cdot)$ and $\Theta_p(\cdot)$ in $\mathcal{L}_{\mathrm{PAT}}$ are implemented as differentiable soft phase estimators based on local peak-response maps, allowing gradients to propagate through the reconstructed waveforms. The PAE is optimized with AdamW using a learning rate of 2×10^{-5} for 120K iterations. The KL weight is set to 5×10^{-5} with a 5K-iteration linear warmup. To encourage cross-modal latent alignment, we use global posterior alignment with weight 5×10^{-5} and a 10K-iteration warmup, cross-modal semantic decodability with weight 5×10^{-4}, and bidirectional InfoNCE instance discrimination with weight 10^{-3}.

C.3 Latent Rectified Flow

We freeze the PAE encoder and decoder and train only the PPG-conditioned latent rectified flow. The target is the ECG latent encoded from the paired ECG segment, and the condition is the PPG latent encoded from the synchronized PPG segment. The rectified-flow model uses a Transformer backbone

with latent channel size 4, temporal length 50, depth 4, hidden dimension 256, and 4 attention heads. During training, Gaussian noise is linearly interpolated with the target ECG latent, and 4 time points are sampled per forward pass. The model is optimized with the full objective in Eq. 26, which combines the rectified-flow loss with the simulator-guided residual losses. The latent rectified flow is trained with Adam using a base learning rate of 1×10^{-4} and betas $(0.9, 0.96)$ for 50K iterations. We use warmup and ReduceLROnPlateau scheduling, with 1K warmup steps, patience 1K, and decay factor 0.5. Gradients are clipped to a maximum norm of 1.0. An exponential moving average model is maintained with decay 0.995 and updated every 10 steps.

C.4 Evaluation Metrics

All metrics compare each generated ECG window $\hat{\mathbf{x}}_e^{(n)}$ with its paired real ECG window $\mathbf{x}_e^{(n)}$, $n = 1, \ldots, N_{\text{test}}$, where each window contains $d = 1200$ samples (10 s at 120 Hz). The main experiments use per-window z-score normalization (Appendix C.1).

Point-wise fidelity. Mean absolute error (MAE) and root mean squared error (RMSE) measure sample-level agreement between paired windows. Both are lower when the reconstructed amplitudes are closer to the reference; RMSE gives larger errors more weight:

$$\text{MAE} = \frac{1}{N_{\text{test}}\, d} \sum_{n=1}^{N_{\text{test}}} \sum_{t=1}^{d} \left| \hat{x}_{e,t}^{(n)} - x_{e,t}^{(n)} \right|, \qquad \text{RMSE} = \frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} \sqrt{\frac{1}{d} \sum_{t=1}^{d} \left(\hat{x}_{e,t}^{(n)} - x_{e,t}^{(n)} \right)^2}. \tag{51}$$

Distribution-level fidelity. FD and FID compare the distribution of generated ECGs with that of real ECGs rather than individual pairs. Given a feature map ϕ, let $(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r)$ and $(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ denote the empirical mean and covariance of $\phi(\mathbf{x}_e)$ over real windows and of $\phi(\hat{\mathbf{x}}_e)$ over generated windows. The Fréchet distance between the two Gaussian approximations is

$$\mathcal{F}_\phi = \left\| \boldsymbol{\mu}_r - \boldsymbol{\mu}_g \right\|_2^2 + \text{Tr}\left(\boldsymbol{\Sigma}_r + \boldsymbol{\Sigma}_g - 2\left(\boldsymbol{\Sigma}_r \boldsymbol{\Sigma}_g \right)^{1/2} \right). \tag{52}$$

FD evaluates Eq. 52 directly in signal space, with ϕ the identity map on the normalized waveform. FID [20] evaluates Eq. 52 in the embedding space of a pretrained ECG encoder, following SE-Diff [56], and thus measures the distributional discrepancy in a learned ECG feature space.

Heart-rate consistency. R-peaks are detected in both the real and generated ECG with NeuroKit2 [36]. For a window whose mean RR interval is $\overline{\text{RR}}$ seconds, the heart rate is $\text{HR} = 60/\overline{\text{RR}}$ beats per minute, and

$$\text{HR MAE} = \frac{1}{N_{\text{test}}} \sum_{n=1}^{N_{\text{test}}} \left| \text{HR}\big(\hat{\mathbf{x}}_e^{(n)}\big) - \text{HR}\big(\mathbf{x}_e^{(n)}\big) \right|, \tag{53}$$

reported in beats per minute.

Waveform correlation (Pearson r). We evaluate the last two columns of Table 1 on 10,254 MC-MED test windows without shifting either signal in time. For each generated window $\hat{x}$ and reference window x, let $u = \hat{x} - \overline{\hat{x}}$ and $v = x - \bar{x}$ denote the mean-centered waveforms. Pearson $r = \langle u, v \rangle / (\|u\|_2 \|v\|_2)$ measures whether amplitudes rise and fall together at corresponding samples. Values range from -1 to 1, with higher values indicating closer agreement in waveform shape.

Beat timing (R-peak F1). In the same windows, we match detected R-peaks in the generated ECG to reference R-peaks within 50 ms (six samples at 120 Hz), allowing each peak to be matched at most once. Matched pairs count as true positives (TP), unmatched generated peaks as false positives (FP), and missed reference peaks as false negatives (FN). The score $\text{F1} = 2\text{TP}/(2\text{TP} + \text{FP} + \text{FN})$ ranges from 0 to 1, with 1 indicating that every generated and reference peak was matched.

Morphology and interval fidelity. Fiducial points are obtained with the discrete-wavelet-transform delineation of NeuroKit2 [36], which locates the onset and offset of the P wave, the QRS complex, and the T wave around each detected R-peak. From these points, we compute the following measurements for each beat:

- *PR interval*: from P-wave onset to QRS onset;
- *QRS duration*: from QRS onset to QRS offset (the J point);
- *QT interval*: from QRS onset to T-wave offset;
- *QTcF*: the Fridericia-corrected QT interval, $\mathrm{QTcF} = \mathrm{QT}/\mathrm{RR}^{1/3}$, with RR in seconds;
- *P-wave and T-wave duration*: from onset to offset of the corresponding wave;
- *ST-J60 deviation*: the amplitude 60 ms after the J point relative to the isoelectric baseline, in normalized amplitude units.

Each measurement is averaged over the beats of a window, and Table 3 reports its MAE between generated and real ECGs over test windows. Interval and duration errors are reported in milliseconds.

D Additional Results

D.1 Sampling Steps and Inference Time

We vary the number of Euler steps used to generate an ECG from a PPG window on MC-MED (Table 5). The main evaluation uses 10 steps. Among the four settings tested, five steps give the lowest waveform MAE and RMSE and take 2.47 ms per window, compared with 4.42 ms at ten steps. Heart-rate MAE varies by only 0.2 bpm across the four settings.

Table 5: Generation error and inference time on MC-MED. Time per 10-second ECG is amortized over a batch of 64 windows and includes PPG encoding, flow integration, and ECG decoding.

Euler steps	MAE	RMSE	HR MAE (bpm)	Time per ECG (ms)
5	0.66	1.02	3.9	2.47
10	0.71	1.07	4.0	4.42
15	0.73	1.09	4.0	6.39
20	0.74	1.10	4.1	8.40

D.2 Sensitivity to Signal Normalization

The main experiments use per-window z-score normalization. Table 6 compares this choice with min–max normalization on MC-MED. MAE and RMSE depend on the scale of the normalized waveform, and the same change in input scale can affect FD and FID. The timing errors use common units across both settings. Heart-rate MAE differs by 0.01 bpm, and each interval or duration MAE differs by at most 0.7 ms.

Table 6: Effect of normalization on MC-MED. HR MAE is in bpm. Interval and duration errors are MAEs in milliseconds.

Normalization	MAE	RMSE	FD	FID	HR MAE
Min–max	0.57	0.77	21.27	16.53	3.93
Z-score	0.71	1.07	39.68	37.79	3.94

Normalization	PR	QRS	QT	QTcF	P duration	T duration
Min–max	59.2	39.0	29.9	32.2	19.7	21.0
Z-score	59.9	38.6	29.3	32.2	19.3	21.3

E Limitations

We evaluate waveform and heart-rate errors on three external datasets. The morphology, interval, and cardiovascular disease classification results are limited to MC-MED, so their generalization to other clinical settings remains untested.

F Broader impacts

PPG2ECG may have positive impacts for scalable cardiovascular monitoring. Since PPG is widely available in wearable devices and bedside monitors, generating ECG-like signals from PPG could support research on long-term cardiac monitoring, data augmentation, and retrospective analysis when ECG is unavailable or difficult to collect continuously. The proposed physiology-guided framework may also encourage more clinically grounded generative modeling for physiological time series.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction match the proposed method and experimental evaluations.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses dataset scope, simulator assumptions, signal quality, and clinical deployment limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper provides the assumptions and proof sketch for the Euler-residual consistency result in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the data, preprocessing, model, objectives, metrics, and implementation details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The clinical dataset (MC-MED) is public dataset, and the current draft provides fully open code.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The paper reports preprocessing, splits, hyperparameters, optimizer, model settings, and evaluation metrics.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Table 1 reports across-run standard deviations.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports GPU type, memory, runtime, number of runs, and total compute.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses an existing clinical dataset under its access terms and does not attempt patient re-identification.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses benefits for wearable monitoring and risks such as misuse, bias, and inappropriate clinical use.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release high-risk datasets, large pretrained models, or clinical deployment systems.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites existing assets and describes their available licenses or access terms.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released code is documented with setup, preprocessing, training, and evaluation instructions.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or new human-subject experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not collect new patient data and uses an existing clinical dataset.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not used as part of the core method; any use was limited to writing assistance.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.