Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

SIR-PhysNet: a deadline-aware ultra-lightweight pipeline for camera-based heart-rate monitoring on embedded hardware

Haochen Chai
Northeastern University
Zining Liu
Northeastern University
Wentao Zhang
Northeastern University
Fangfang Jiang
jiangff@bmie.neu.edu.cn
Northeastern University

Research Article

Keywords: Remote photoplethysmography (rPPG), Realtime image processing, Pipeline profiling, Detect-and-track scheduling, Heart-rate monitoring, Dynamic Fourier filtering

Posted Date: September 8th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10816153/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

SIR-PhysNet: a deadline-aware ultra-lightweight pipeline for camera-based heart-rate monitoring on embedded hardware

Haochen Chai · Zining Liu · Wentao Zhang · Fangfang Jiang

Abstract Camera-based heart rate (HR) monitoring on an edge device is limited as much by the frame budget of the complete pipeline as by the cost of network inference. A monitor that must return a new estimate every 2.5 s also has to decode video, track a face and standardize the skin region on the same four cores, and a parameter count does not indicate whether they fit. We present SIR-PhysNet, a remote photoplethysmography (rPPG) pipeline designed around that budget. Geometric, skin-region and illumination variation is handled by fixed-cost image operations in front of the network rather than learned inside it, and the rate is read out through an analytical Fourier mask rather than a dense waveform decoder, so the learned component requires only 0.26 M parameters and 0.12 GFLOPs. Under subject-disjoint protocols the method reaches 2.75 BPM RMSE with $r = 0.98$ on UBFC-rPPG and 3.39 BPM MAE on the motion-intensive ECG-Fitness set. Profiled on a fan-cooled Raspberry Pi 5, its streaming front end sustains 34.60 fps, remains above the 30-fps input rate through an eight-hour run with no throttling flag raised, and spends approximately 70% of each frame interval on face detection and ROI rectification. Clip-level network cost is reported as a parameter-matched allowance rather than as trained-graph latency. Face processing, rather than network inference, limits the achievable frame rate on this class of hardware.

Keywords Remote photoplethysmography (rPPG) · Real-time image processing · Pipeline profiling · Detect-and-track scheduling · Heart-rate monitoring · Dynamic Fourier filtering

H. Chai · Z. Liu · W. Zhang · F. Jiang
College of Medicine and Biological Information Engineering, Northeastern University, Shenyang 110179, Liaoning, China
E-mail: jiangff@bmie.neu.edu.cn
Corresponding author: Fangfang Jiang

1 Introduction

Real-time operation is a property of a complete pipeline rather than of any single stage within it. In the configuration used throughout this paper, video arrives at 30 fps, the analysis window is 150 frames and the stride is 75 frames, so a new heart rate (HR) estimate is due every 2.5 s. Within that interval the system must also decode 75 frames, locate and rectify a face in each of them, and standardize the skin region. Parameter and operation counts describe none of these requirements: a model may be small in FLOPs and still miss the deadline, or meet it for ten minutes and fail after six hours of continuous load. Establishing real-time behaviour therefore requires measuring the assembled pipeline on the target device, stage by stage, over a duration comparable to the intended use.

Remote photoplethysmography (rPPG) recovers the cardiac rhythm from the small periodic colour changes that blood volume induces in facial skin [28, 36], and cameras are attractive sensors for it because they are already installed in intelligent cabins, kiosks, telehealth terminals and driver monitoring systems [2, 4, 10]. A deployable monitor is nonetheless a chain of image-processing stages rather than a network with a thin wrapper: decoding, face detection and tracking, mapping to a canonical region of interest, and attenuation of illumination drift and non-skin pixels all precede the temporal model. All of these stages draw on the same time budget. In the measurements reported here, detection and rectification alone account for approximately 70% of the profiled per-frame preprocessing time and are an order of magnitude more expensive than decoding, a distribution that no FLOP table reports.

The chain is expensive because the input signal is weak. The pulsatile component of skin colour is on the order of one percent of the pixel value, and head motion, expressions, illumination change, auto-exposure, compression artefacts and mechanical vibration can all exceed it [23, 24, 40]. Robust-

ness against these factors can be learned inside the network or imposed in front of it by deterministic image operations. The first approach is conventional but expensive, since invariance is paid for in capacity, in training data and, at run time, in operations on every frame. The second is inexpensive and, more important here, predictable: an affine map onto a canonical face, a skin mask derived from landmarks and an illumination quotient computed by a guided filter require a fixed few milliseconds per frame and do not grow with model size. SIR-PhysNet takes the second approach, and standardizing the input in this way allows its learned component to remain small.

The second design choice concerns the quantity the model must emit. Most deep rPPG models regress a dense blood volume pulse (BVP) waveform and read the rate from its spectrum, committing a large share of their capacity to morphology that a rate-only monitor never uses [5, 45]. Predicting the dominant cardiac frequency directly and deriving a waveform from it analytically is more economical. The **HR-guided dynamic Fourier filter (HR-DFF)** does this: a Gaussian mask centred on the predicted frequency is applied to the temporal tokens in the Fourier domain, and the surrogate signal is the inverse transform. The mask is computed rather than learned, so the readout carries no trainable parameters.

This paper presents **SIR-PhysNet**, the pipeline that follows from these two choices, together with measurements characterizing its behaviour as a real-time system. The contributions are fourfold. (i) We treat edge rPPG as an image-processing task with an explicit deadline and organize the pipeline around that budget, moving nuisance invariance into fixed-cost preprocessing so that the learned model remains at 0.26 M parameters and 0.12 GFLOPs. (ii) We introduce HR-DFF, a readout that separates rate estimation from waveform generation through an analytical Fourier mask conditioned on a learned HR prior; since the mask is evaluated rather than fitted, the readout contributes no trainable parameters. We present this as an argument about parameter economy and claim no independent accuracy advantage over a comparably sized direct HR head. (iii) We report subject-disjoint accuracy with dataset-appropriate supervision on UBFC-rPPG, MAHNOB-HCI and ECG-Fitness [3, 30, 32], separating protocol-matched comparisons from contextual literature values throughout. (iv) We profile the streaming front end on a fan-cooled Raspberry Pi 5 with per-stage timing, two front-end scheduling policies and an eight-hour soak test, and quantify the distribution of the per-frame budget. Face detection and ROI rectification, rather than the network, set the achievable frame rate.

2 Related work

2.1 Establishing real-time behaviour by measurement

Work in this and adjacent venues establishes real-time behaviour by measurement on the target platform rather than by operation counts. Recent examples include power-aware adaptive-rate tracking on the Jetson Nano [8], hardware and software partitioning of an imaging chain on an SoC FPGA [27], and host-side multithreading analysis on the Kria KV260 [9]. These studies state a deadline, break the cost down per stage, and take resource telemetry from the deployed configuration rather than from a desktop proxy; Sect. 5 adopts the same practice. Within rPPG, RhythmEdge [13] is the closest precedent, reporting device-level latency, memory and power for an end-to-end contactless HR system. The present study differs in profiling the front-end stages separately and in stating which part of the network cost is measured and which is an allowance.

2.2 Face front ends and detect-and-track scheduling

The front end of any face-based monitor consists of a detector and a tracker. MediaPipe Face Mesh [15, 22] supplies dense landmarks at a cost that is modest on a desktop but substantial on a quad-core embedded CPU, which is why detect-and-track scheduling, with the detector running periodically and optical flow propagating the region between detections, is standard in embedded pipelines. rPPG studies rarely budget for this stage, although it dominates the per-frame cost measured here and affects throughput more than any architectural decision available at this parameter scale.

2.3 Classical and learned rPPG estimation

Classical rPPG estimates the pulse from channel statistics or chrominance projections [12, 28, 36, 38]. These methods are almost free computationally and remain useful baselines, but degrade rapidly under head motion and uncontrolled illumination. Reproducibility studies and standardized toolkits have argued that geometry, frame rate and input quality belong to the method rather than to the experimental setup [21, 31], which is the position adopted here. Learned models improved robustness by assigning that responsibility to the network: DeepPhys [6] and PhysNet [43] established the spatiotemporal formulation, EfficientPhys [20] removed part of the preprocessing in exchange for a larger backbone, and PhysFormer [45] improved waveform quality with a markedly larger budget, later extended by a slow-fast temporal-difference design [44]. LSTC-rPPG [17] pairs long-term and short-term convolutional paths. All pursue

efficiency through temporal-architecture design rather than through input standardization.

2.4 Efficiency-oriented designs and our position

Newer designs seek efficiency through state-space temporal modelling [46], codebook lookup of clean pulse segments [7], frequency-domain losses with multi-scale fusion [14], label-efficient contrastive training [33] and memory-efficient edge inference [37]. Vance and Flynn [35] use representational-similarity analysis to show how much rPPG capacity is redundant, the empirical counterpart of shrinking the learned component once nuisance handling moves upstream. Nearly all of this work reports parameters and FLOPs, few report what those translate into on a named device, and fewer still report the cost of the surrounding stages. SIR-PhysNet differs in where the effort is placed: invariance is imposed before the network, the waveform is produced by a parameter-free spectral operation afterwards, and the pipeline is profiled on hardware.

3 Methods

The design follows two rules. Nuisance invariance is imposed by fixed-cost image operations in front of the network rather than learned inside it, which keeps the per-frame cost predictable and the model small; and the readout produces the rate that the application consumes, with the waveform derived from it analytically, which removes the dense decoder that usually dominates parameter count. The subsections below describe the pipeline in processing order.

3.1 Pipeline and windowing

Figure 1 summarizes the processing flow. At the default $f_s = 30\,\mathrm{Hz}$, clips contain $T = 150$ frames with stride $S = 75$ (50% overlap), while ECG-Fitness uses $T = 300$ to provide a 10 s context under high motion. Short clips are replicated at their boundaries and accompanied by a binary validity mask, so padded positions do not contribute to framewise losses. Stages 1 to 3 execute once per frame; network inference with the spectral readout executes once per 75-frame stride.

3.2 ROI stabilization and quality check

MediaPipe Face Mesh [15, 22] provides 468 landmarks. A partial affine transform uses landmark indices 33, 133, 263, 362, 1, 4, 9, 61 and 291 to map each face to a 200×200 canonical ROI. A result is rejected when its implied in-plane rotation exceeds 15° or its inter-frame centre displacement exceeds 10 pixels, and the most recent valid ROI is then reused to preserve temporal continuity. The hybrid schedule of Sect. 5 applies optical-flow tracking between periodic Face Mesh updates without changing this canonical interface.

3.3 Signal-informed standardization

3.3.1 Skin support and illumination suppression

The landmark convex hull defines a skin support after eye and lip polygons are removed. A fixed soft prior $\mathbf{W}_{\mathrm{ppg}}$ emphasizes forehead and cheek regions, and Gaussian smoothing with contrast $\kappa = 2$ produces $\mathbf{M} = \mathbf{M}_{\mathrm{skin}} \odot \sigma(\kappa \mathbf{G} * \mathbf{W}_{\mathrm{ppg}})$. To suppress illumination drift, a guided filter of radius 15 estimates the smooth intensity component $\mathbf{L}_t$. The illumination quotient is

$$\mathbf{R}_t = \frac{\mathbf{I}_t^{\mathrm{rgb}}}{\mathbf{L}_t + \varepsilon}, \tag{1}$$

and the standardized stream is obtained by weighted fusion,

$$\mathbf{I}_t^{\mathrm{ill}} = 0.5\,\mathbf{I}_t^{\mathrm{rgb}} + 0.5\,\mathbf{R}_t. \tag{2}$$

3.3.2 Two-stream construction, masking and resizing

The same mask is applied to both streams, which are resized and channel-concatenated as

$$\mathbf{X}_t = \mathrm{concat}\Big(\mathrm{resize}(\mathbf{I}_t^{\mathrm{rgb}} \odot \mathbf{M}),\ \mathrm{resize}(\mathbf{I}_t^{\mathrm{ill}} \odot \mathbf{M})\Big), \tag{3}$$

which yields $C = 6$ channels.

3.4 Grid-based spatiotemporal features

Each normalized 320×240 frame is partitioned into an 8×8 grid. The mean RGB value in block b is $\mu_{b,c}(t) = |\Omega_b|^{-1} \sum_{(x,y)\in\Omega_b} I_c(x, y, t)$, which produces $\boldsymbol{f}_t \in \mathbb{R}^{192}$. Stacked descriptors form $\mathbf{F} \in \mathbb{R}^{T\times 192}$. Online inference maintains this representation in memory and bypasses diagnostic disk export.

3.5 Proposed network

SIR-PhysNet is a compact spatiotemporal model for clip-level rPPG estimation (Fig. 2). Because the input arrives already stabilized, masked and illumination-normalized, its capacity is allocated to pulse dynamics rather than to nuisance invariance.

3.5.1 Input interface and early encoding

Each clip has shape $\mathbf{X} \in \mathbb{R}^{B\times T\times 6\times 16\times 16}$. Grid descriptors are reshaped to 8×8 and upsampled to this interface, and

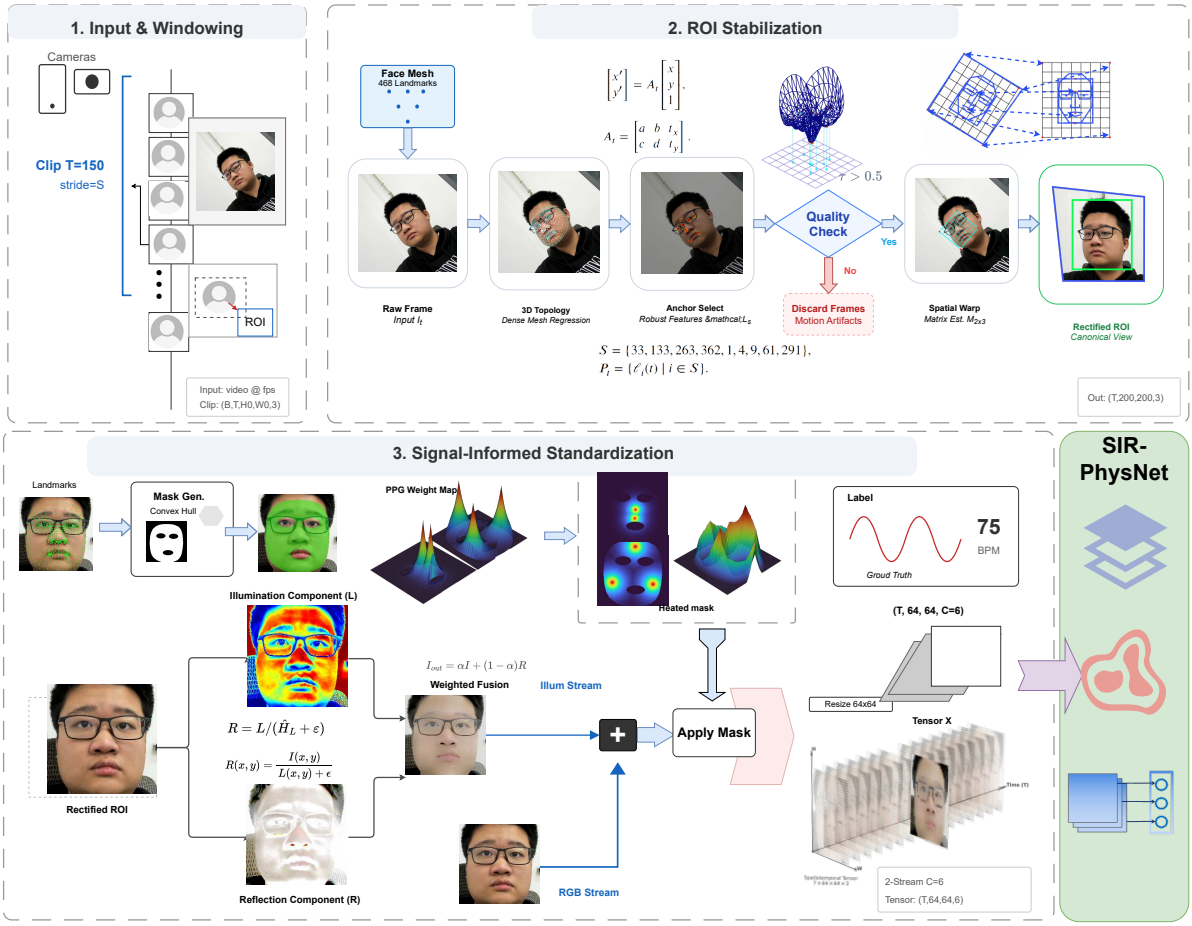


Fig. 1 End-to-end SIR-PhysNet processing flow. Video windowing and face-based ROI stabilization precede skin masking, illumination suppression, two-stream construction and compact network inference. Stages 1–3 execute once per frame; network inference with the spectral readout executes once per 75-frame stride. The facial frames are illustrative material recorded by one of the authors.

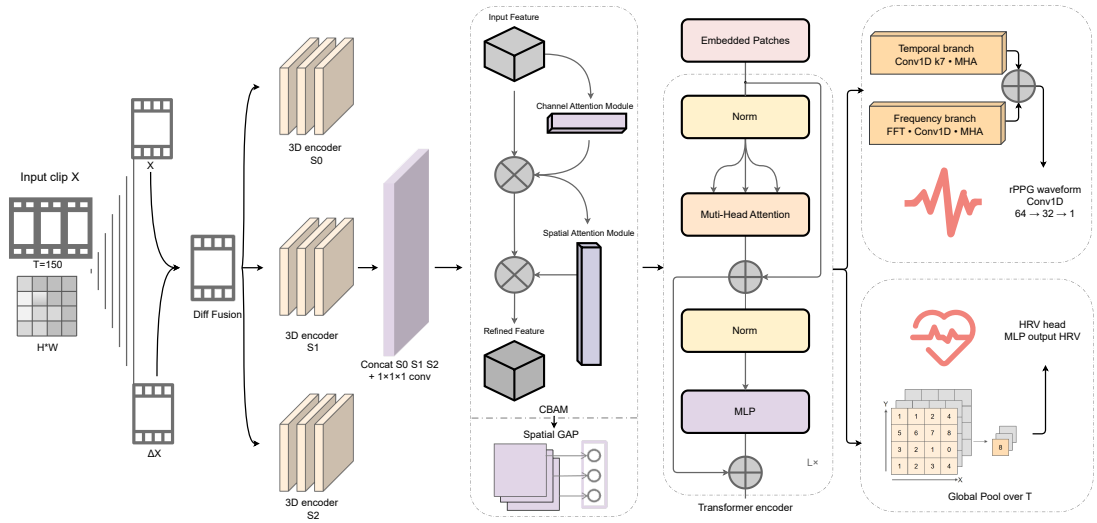


Fig. 2 Architecture of SIR-PhysNet. The aligned input clip **X** is fused and encoded by multi-scale 3D encoders. Rather than regressing a dense waveform directly, the temporal tokens undergo a time–frequency decoupled readout in which an HR prior guides spectral filtering to align output energy within the physiological band.

normalization uses training-split statistics only. Original and temporal-difference streams are fused by shallow 3-D processing and a $1 \times 1 \times 1$ bottleneck.

3.5.2 Multi-scale features and temporal modelling

Narrow 3-D encoders process Gaussian-pyramid scales 1, 1/2 and 1/4. Upsampled features are concatenated and compressed to $\mathbf{F} \in \mathbb{R}^{B\times 52\times T\times H'\times W'}$ without temporal downsampling. Per-frame CBAM [39] refinement precedes spatial pooling and produces $\mathbf{S} \in \mathbb{R}^{B\times T\times 52}$ for a four-layer temporal Transformer.

3.5.3 Time–frequency decoupled readout

A temporal pooling branch predicts the dominant prior f_{hr} directly. HR-DFF then transforms the tokens to $\mathbf{Z}(f) = \mathcal{F}(\mathbf{S})$ and multiplies them by a Gaussian mask centred on that prediction,

$$\mathbf{M}(f) = \exp\left[-\frac{(f - f_{\text{hr}})^2}{2\sigma^2}\right], \tag{4}$$

and the surrogate signal is the inverse transform of the masked spectrum,

$$\hat{\mathbf{s}} = \mathbf{W}_{\text{out}}\,\mathcal{F}^{-1}(\mathbf{Z} \odot \mathbf{M}) + \mathbf{b}_{\text{out}}. \tag{5}$$

The mask is evaluated from f_{hr} rather than fitted, so the FFT, the masking and the inverse FFT add no learnable parameters; only the output projection does. The complete 5-s model has 0.26 M parameters and 0.12 GFLOPs.

3.5.4 Training objective and configuration

For prediction $\hat{\mathbf{s}}$ and dataset-specific reference $\mathbf{s}$, training uses

$$\mathcal{L} = \alpha\mathcal{L}_{\text{MSE}} + \beta\mathcal{L}_{\text{PSD}} + \gamma(1 - \rho), \tag{6}$$

where $\mathcal{L}_{\text{PSD}}$ is the L_1 distance between normalized spectra over 0.67–3.33 Hz. UBFC-rPPG and MAHNOB-HCI use $(\alpha, \beta, \gamma) = (1, 0.05, 0.1)$ with contact PPG and smoothed ECG-derived pseudo-BVP respectively. ECG-Fitness uses raw ECG only for rhythm and frequency alignment and ECG-derived HR as the primary anchor, and $(0.2, 0.05, 0)$ reduces morphological mismatch.

The PyTorch model is trained with AdamW, a learning rate of 1.2×10^{-4}, weight decay of 5×10^{-5} and batch size 4, with a plateau-triggered learning-rate scheduler and gradients clipped at 2.0. Augmentation comprises amplitude scaling $\mathcal{U}[0.8, 1.2]$, circular shifts of ±10 frames, Gaussian noise, and monotone time warping, which is disabled for high-motion data. All normalization statistics are computed on the training split.

4 Experiments and results

4.1 Datasets and supervision

UBFC-rPPG contains 42 subjects recorded at 640×480 and 30 fps with synchronized contact PPG [3], which is used in the full loss of Eq. 6. The MAHNOB-HCI tagging subset contains 527 videos from 27 subjects with expression and natural head motion [30]; it is supervised by smoothed ECG-derived pseudo-BVP and only HR metrics are reported. ECG-Fitness contains 207 treadmill-exercise videos from 17 subjects and is the motion-intensive stress test used here [32]. It provides no contact PPG, and because ECG and peripheral PPG share rate but differ in phase and morphology [11], ECG-derived HR is the primary anchor and raw ECG is used only for frequency alignment. Waveform-morphology metrics are not reported for that dataset.

4.2 Protocol, baselines and metrics

UBFC-rPPG and MAHNOB-HCI follow the subject-disjoint rPPG-Toolbox protocol [21], in which all clips from one subject remain in one fold. ECG-Fitness follows the same rule, with its 17 subjects split across folds and no subject contributing clips to more than one fold. Each dataset uses 4-fold cross-validation, and every learned configuration is trained under three random seeds, 1005, 2026 and 2004, on the same fold assignment. Splits, seeds and training-only normalization are fixed, and the accuracy experiments are independent of the embedded profiling. Reported values are seed means; the individual runs were not retained, so no per-seed intervals are given. Overlapping clips drawn from one subject are also not independent observations, so comparisons whose margins are small relative to the between-subject spread should be read accordingly.

Baselines comprise Green [36], ICA [28], CHROM [12], POS [38], SAMC [34], DeepPhys [6], HR-CNN [32] and PhysNet/STVEN+rPPGNet [43], evaluated using the unified protocol. All waveform-producing methods use zero-phase Butterworth filtering over 40–200 BPM followed by one-sided PSD peak picking over 0.67–3.33 Hz, with $\hat{h} = 60 f^{\star}$. SD, MAE and RMSE follow from the per-clip error $e_i = \hat{h}_i - h_i$, and r is the Pearson correlation between predicted and reference rates over all clips.

4.3 UBFC-rPPG benchmark

Table 1 reports results on UBFC-rPPG, where every listed baseline was reproduced under the same protocol. The full pipeline gives the lowest SD (2.76 BPM) and RMSE (2.75 BPM) and the highest correlation (0.98) among the reproduced methods, with an MAE of 2.04 BPM against

Table 1 HR estimation on UBFC-rPPG. All baselines were reproduced by us under the unified rPPG-Toolbox subject-disjoint protocol with identical splits, windows and reference-HR extraction, so this is a controlled comparison. Values are means over seeds 1005, 2026 and 2004.

Method	SD↓	MAE↓	RMSE↓	r↑
Classical				
Green [36]	13.5	7.52	14.4	0.58
ICA [28]	10.8	5.20	11.5	0.62
CHROM [12]	5.10	2.37	5.61	0.90
POS [38]	7.25	3.13	7.55	0.86
Deep Learning				
DeepPhys [6]	4.75	2.42	4.29	0.93
PhysNet [43]	3.81	**1.81**	3.51	0.96
Ours (Full)	**2.76**	2.04	**2.75**	**0.98**

1.81 BPM for PhysNet. These margins are small relative to the between-subject spread on this dataset, so the result places SIR-PhysNet in the same accuracy band as PhysNet and DeepPhys at a fraction of their parameter count rather than establishing a ranking. UBFC-rPPG is a controlled recording condition, and the motion-intensive setting of Sect. 4.5 provides the more informative test.

4.4 MAHNOB-HCI tagging

On MAHNOB-HCI Tagging the proposed method achieves 3.71 BPM MAE, 5.70 BPM RMSE and $r = 0.92$ (Table 2). The correlation is the highest listed, against 0.85 for Meta-rPPG [16], while the MAE is not the lowest. Because the baseline values are drawn from prior publications, the table is contextual rather than a protocol-matched ranking.

4.5 Motion-robust evaluation on ECG-Fitness

ECG-Fitness [32] combines treadmill-induced head and body motion with breathing displacement and illumination variation. In Table 3, POS [38] and DeepPhys [6] exceed 10 BPM MAE, whereas EfficientPhys [20] and BigSmall [25] reduce the error substantially. SIR-PhysNet reaches 3.39 BPM MAE and 4.80 BPM RMSE with 0.26 M parameters and 0.12 GFLOPs. No claim of superiority over BigSmall or EfficientPhys is made, because their values are literature-reported and differences in split construction, window length, preprocessing and reference-HR extraction can move rPPG scores by more than the margins at issue. The table supports a narrower statement: a model with 0.26 M parameters operating on grid descriptors falls within the same error range as recent methods an order of magnitude larger, which is the observation on which the deadline argument depends.

Table 2 HR estimation on MAHNOB-HCI Tagging. Contextual comparison, not a controlled ranking: all baseline values are literature-reported under each source's own split, preprocessing, window length and reference-HR construction.

Method	SD↓	MAE↓	RMSE↓	r↑
Classical				
Poh et al. [28]	13.5	—	13.6	0.36
Li et al. [18]	6.88	—	7.62	0.81
CHROM [12]	—	13.49	22.36	0.21
SAMC [34]	5.81	—	6.23	0.83
Deep Learning				
SynRhythm [26]	10.88	—	11.08	—
HR-CNN [32]	—	7.25	9.24	0.51
DeepPhys† [6]	—	4.57	—	—
PhysNet [43]	7.84	5.96	7.88	0.76
AutoHR† [41]	4.73	3.78	5.10	0.86
PhysFormer† [45]	**3.87**	—	—	0.87
Meta-rPPG [16]	4.90	**3.01**	**3.68**	0.85
Ours (Full)	5.69	3.71	5.70	**0.92**

Baselines sourced from Meta-rPPG [16]. †Values cited from [42, 45]; cross-publication comparisons are contextual.

Table 3 HR estimation on ECG-Fitness. Contextual comparison: baseline values are literature-reported under their own protocols and are not protocol-matched to ours.

Method	MAE↓	RMSE↓	r↑
Classical			
POS [38]	12.33	19.53	0.53
CHROM [12]	10.60	16.50	0.61
Deep Learning			
DeepPhys [6]	11.69	16.89	0.57
PhysNet [43]	5.89	10.58	0.81
PhysFormer [45]	12.10	—	—
EfficientPhys [20]	4.38	6.78	0.92
BigSmall [25]	3.85	6.21	**0.95**
Ours (Full pipeline)	**3.39**	**4.80**	0.94

Only HR metrics are reported, since waveform-morphology metrics are not applicable under ECG-only supervision. Baselines cited from [20, 25]; PhysFormer MAE from Sect. 4.6. All methods are supervised by ECG-derived HR.

4.6 Accuracy and complexity

Table 4 places the design in the accuracy-efficiency trade-off space. Classical methods require negligible compute but degrade under intense motion, whereas modern high-capacity architectures handle motion artefacts more effectively at 0.23–1.25 GFLOPs, much of which is devoted to reproducing fine-grained waveform detail. Across the listed methods the two axes are only loosely coupled: on ECG-Fitness the highest MAE belongs to the largest model, and on UBFC-rPPG the lowest belongs to one of the smallest. Those accuracy values are literature-reported rather than protocol-matched, so this

is a reason not to use compute as a proxy for accuracy rather than a finding about the individual methods.

At 0.26 M parameters and 0.12 GFLOPs, SIR-PhysNet occupies the low-compute region of Table 4 while reaching 3.39 BPM MAE on ECG-Fitness, and HR-DFF contributes none of those parameters. RhythmMamba [46], CodePhys [7] and LiteSyncNet [14] use different resolutions and clip lengths, so their published complexity figures are not folded into the table. The comparison describes an operating point rather than a runtime ranking on a single device, which is the subject of Sect. 5.

4.7 Component ablation

Table 5 removes one component at a time while holding the backbone and the training schedule fixed. Removing alignment causes the largest change, from 3.39 to 8.53 BPM MAE, followed by motion fusion at 6.70 BPM and the HR-DFF readout at 4.76 BPM, while skin masking and illumination suppression produce smaller but consistent differences. The ordering identifies geometric stabilization and motion handling as the main contributors under treadmill motion, with the spectral readout next. Figure 3 complements the table with regression scatter plots and per-clip error distributions.

5 Real-time performance on embedded hardware

5.1 Platform, workload and measurement boundary

The profiling target was a fan-cooled Raspberry Pi 5 (BCM2712, four Cortex-A76 cores at 2.4 GHz, 8 GB RAM) running the streaming prototype on four CPU threads. Fan-speed and PWM telemetry were not exposed, so cooling is characterized only through CPU temperature and the firmware throttling flags. The input was a prerecorded 1088×720 clip at 28.2839 fps, looped so that sustained load could be measured without camera-capture variability. The pipeline used a 200×200 rectified ROI, an 8×8 grid, 150-frame clips and a 75-frame stride. Two schedules for the face front end were compared: the first invokes MediaPipe Face Mesh on every frame, the second invokes it every fifth frame and propagates the ROI with optical flow in between.

A monotonic per-stage clock times decoding, face and ROI processing, and standardization and grid construction on every frame, and times the forward pass on every completed clip. PyTorch is restricted to four compute threads with a single interop thread, and warm-up forwards precede the measured region. Process CPU share, resident set size, SoC temperature and the `vcgencmd` throttling word are sampled every 10 s by a separate thread. Each row of Table 6 is one completed run summarized over its frame-level and clip-level samples. Quantities absent from the retained archive, including the warm-up count and the latency standard deviation, are left unreported rather than imputed.

Two boundaries apply to all results in this section. First, the experiment measures implementation cost and stability rather than accuracy, and no HR error was computed on the device. Second, the timed network component was a 260,000-parameter scaffold matched to SIR-PhysNet in parameter count but not in computational structure, since most of its parameters sit in a wide fully-connected stage rather than in temporal attention. Its clip time therefore bounds the inference allowance available at this parameter scale and permits no inference about the latency of the trained graph. Everything upstream of the network in Table 6 is the pipeline of Sect. 3 itself.

Processing capacity is $C = N_f / t_{\text{active}}$, where N_f is the number of processed frames and t_{active} the measured processing wall time; its reciprocal gives the active cost per frame, $L_f = 1000/C$ ms. At 30 fps the stride deadline is $D_{\text{stride}} = S/f_s = 75/30 = 2.5$ s. A configuration keeps pace with the nominal input stream when $C \geq 30$ fps, and clip inference must additionally complete within D_{stride}.

5.2 Throughput, latency and the scheduling trade-off

The front end clears the nominal 30-fps requirement under both schedules (Table 6, Fig. 4). Running Face Mesh on every frame gives 34.60 fps at 28.90 ms of active cost per frame, which is the configuration relevant to Sect. 4 since every reported HR figure was obtained under per-frame detection. The detect-and-track variant reaches 46.97 fps at 21.29 ms, a 35.8% gain in throughput for a 26.3% reduction in per-frame cost, with median face and ROI latency falling from 15.69 to 4.43 ms and a valid-face rate of 88.98% against 89.30%. Its P95 remains 19.88 ms because every fifth frame still triggers a full detector call, so median and P95 together describe a bimodal cost.

The tracked schedule is reported as a cost measurement and as evidence that front-end headroom exists, not as a validated operating point. A comparable valid-face rate shows only that an ROI remains available; it does not show that ROIs propagated by optical flow preserve the stabilized skin signal under motion, rotation, occlusion or tracker drift, and Table 5 indicates how much alignment quality matters. Establishing that would require HR evaluation under the tracked schedule on motion-rich video with reference labels, which has not been performed, so the accuracy and deadline claims of this paper rest on the per-frame schedule alone.

Under the every-frame schedule the per-stage medians are 1.63 ms for decoding, 15.69 ms for face and ROI processing, and 5.19 ms for standardization and grid construction. Face processing is therefore about an order of magnitude more expensive than decoding and roughly three times the cost

Table 4 Accuracy–complexity trade-off. All methods profiled under T=150, 16×16 spatial resolution except where noted.

Method	Params (M) ↓	FLOPs (G) ↓	UBFC MAE ↓	Fitness MAE ↓
DeepPhys [6]	2.80	0.13	2.42	11.69
PhysNet [43]	1.16	0.64	1.81	5.89
TS-CAN [19]	0.53	—	1.70	—
EfficientPhys [20]	2.16	0.23	1.41	4.38
BigSmall [25]	2.50	0.58	—	3.85
PhysFormer [45]	5.05	1.25	2.83	12.10
RhythmMamba [46]	0.58	—	0.50	—
SIR-PhysNet (Ours)	**0.26**	**0.12**	2.04	**3.39**

Parameter and FLOP counts were profiled by us at the stated input configuration. Accuracy values other than ours are literature-reported and not protocol-matched, so rows are comparable on the complexity axes but only contextually on the accuracy axes. TS-CAN and RhythmMamba parameters are literature-reported, their FLOPs would require re-profiling at our input configuration, and their UBFC MAE is cited from [46].

Table 5 Ablation study on ECG-Fitness. Impact of removing components.

Variant	MAE ↓	RMSE ↓	r ↑
(a) Full pipeline	**3.3949**	**4.8005**	**0.9407**
(b) w/o Alignment	8.5330	10.6451	0.5883
(c) w/o Skin Mask	3.8428	5.4697	0.9220
(d) w/o Illum. Suppr.	3.6852	5.2103	0.9302
(e) w/o Motion Fusion	6.7011	8.9557	0.7573
(f) w/o HR-DFF	4.7628	6.4214	0.8881

of the entire standardization path, so the detector schedule is the most effective single means of raising throughput on this platform. The inference scaffold occupies 325.23 ms median and 347.39 ms P95 per 150-frame clip under hybrid scheduling, or 13.0% and 13.9% of the 2.5-s stride budget.

5.3 Sustained thermal and memory behaviour

The longest run consumed 814,576 frames, equivalent to 8 h of source video, over 24,401.61 s (6.78 h) of active load. Capacity held at 33.38 fps, within 0.02 fps of the 4-h run and above the 30-fps requirement throughout. The SoC peaked at 68.85 °C, and every sampled firmware state was 0x0, with no current or historical under-voltage, frequency cap, soft temperature limit or throttling event.

The memory trace requires a more careful account. During active profiling the process RSS rose from 334.27 MiB to a profiler-inclusive peak of 1,226.86 MiB, then fell to 621.48 MiB at the final post-cleanup sample. The archive records the export of a $T \times 192$ grid-feature array; stored as float32, its 814,576 rows require 596.61 MiB, close to the observed cleanup drop of 605.38 MiB. The same agreement holds in the shorter runs, where the observed drop exceeds the expected buffer by 8.63 to 9.19 MiB. Much of the monotonic rise is therefore consistent with retained diagnostic data rather than with the bounded streaming state. The original profiler source was not retained, so this is a quantitative consistency check rather than object-level proof, and the remaining 287.22 MiB of post-cleanup increase cannot be attributed. The peak is reported as an instrumentation-inclusive upper bound and not as a deployment working set; a bounded-memory measurement would require a rerun that streams its logs, disables full-run feature retention and holds only a 150-frame ring buffer. Energy per update is not reported, since no synchronized calibrated power log was kept.

6 Discussion

The embedded measurements identify the face front end, rather than the network, as the stage that sets the achievable frame rate. It consumes approximately 70% of the profiled per-frame budget, and relaxing its schedule raises throughput from 34.60 to 46.97 fps, a larger change than any architectural modification available at this parameter scale. Whether that headroom can be used remains open, since the tracked schedule was characterized by cost and valid-face rate rather than by HR error.

The second effect that an operation count does not predict is the behaviour of the pipeline under sustained load. Capacity in the eight-hour run holds within 0.02 fps of the four-hour run, the SoC peaks at 68.85 °C, and the firmware raises no throttling flag at any sample. These values are measured across 814,576 frames rather than extrapolated from the ten-minute case.

The ablation ordering is consistent with the cost distribution measured on the device. Removing geometric alignment produces the largest degradation, from 3.39 to 8.53 BPM MAE, and alignment also dominates the per-frame budget in Table 6. The stage contributing most to accuracy under treadmill motion is therefore the stage that constrains throughput, so the schedule applied to it, rather than the size of the network, defines the operating point available on this platform.

The supervision available on ECG-Fitness constrains the quantity the model can reasonably be asked to emit. ECG fixes the rate and the fundamental frequency but not the peripheral pulse shape, since the two differ in phase and morphology

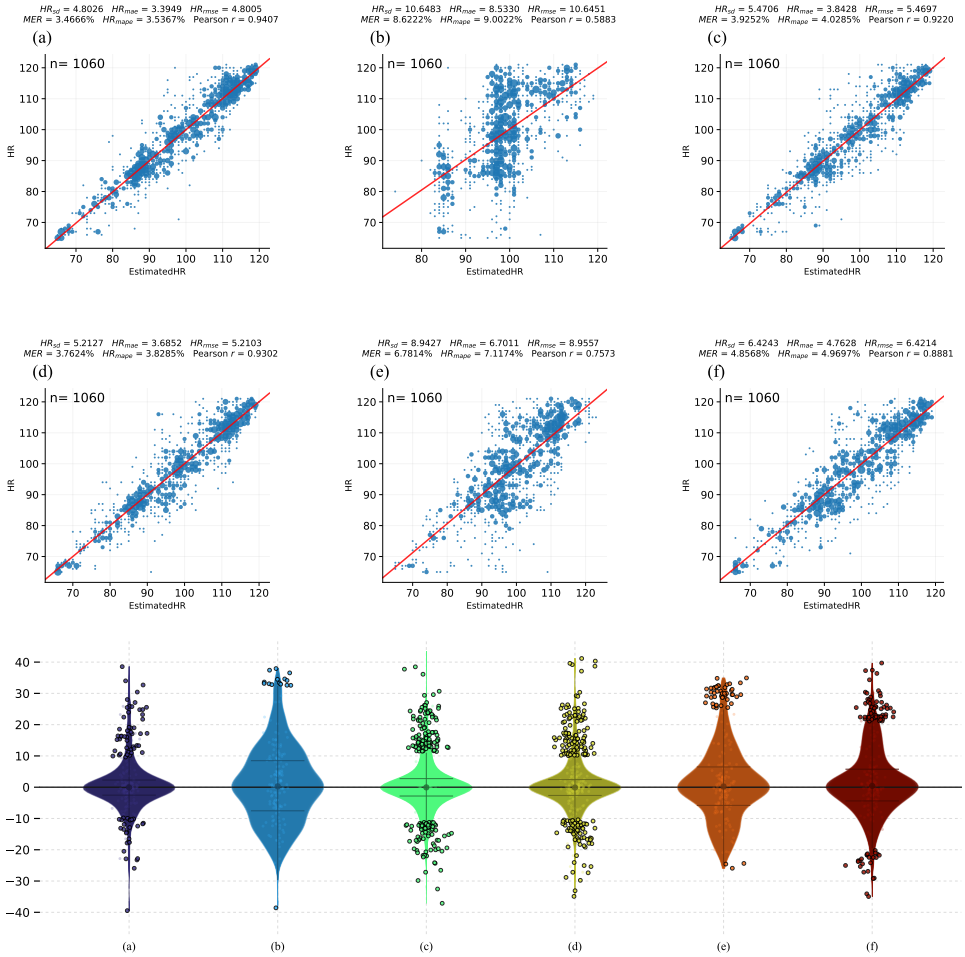


Fig. 3 Visual ablation analysis on ECG-Fitness. Upper panel: regression scatter of predicted versus reference HR for the full pipeline and each component-removal variant, with unified axis limits. Lower panel: per-clip HR error distributions, where violin widths are kernel density estimates and overlaid markers are individual clips. The $n = 1060$ observations are overlapping clips drawn from 17 subjects and are not independent samples, so the panels describe error-distribution shape rather than effective sample size. The full pipeline yields a tighter regression trend and narrower error tails.

Table 6 Raspberry Pi 5 real-time profiling. Latency pairs are median/P95. Source load denotes video content processed, whereas wall time is the measured active processing duration. Network values are parameter-matched inference proxies, not trained SIR-PhysNet latency. Profiler RSS is the instrumentation-inclusive peak and the final post-cleanup sample in MiB, and is not a deployment working set.

Schedule	Source load	Wall time	Capacity (fps)	Active cost (ms/frame)	Face ROI (ms)	Proxy/clip (ms)	Profiler RSS peak/final (MiB)	$T_{\max}$ (°C)	Throttle
Every-frame	10 min	8.17 min	34.60	28.90	15.69/17.59	324.29/348.65	485.39 / 464.33	63.90	No
Hybrid (i=5)	10 min	6.02 min	46.97	21.29	4.43/19.88	325.23/347.39	477.09 / 455.89	63.35	No
Every-frame	4 h	3.39 h	33.36	29.98	16.13/18.24	325.47/348.67	868.08 / 560.58	68.85	No
Every-frame	8 h	6.78 h	33.38	29.96	16.12/18.20	326.17/349.16	1,226.86 / 621.48	68.85	No

through the pre-ejection period and pulse transit time [11], and pulse-rate variability does not reproduce ECG-derived HRV at beat level [1]. The SIR-PhysNet output is accordingly a spectrally aligned surrogate for HR extraction, neither a BVP reconstruction nor an input to any beat-level decision.

The efficiency reported here originates in an explicit spectral construction rather than in a larger learned representation. HR-DFF applies an analytical Fourier mask conditioned on a learned HR prior and contributes no parameters; that choice, together with the front-end schedule, defines the operating point reported here.

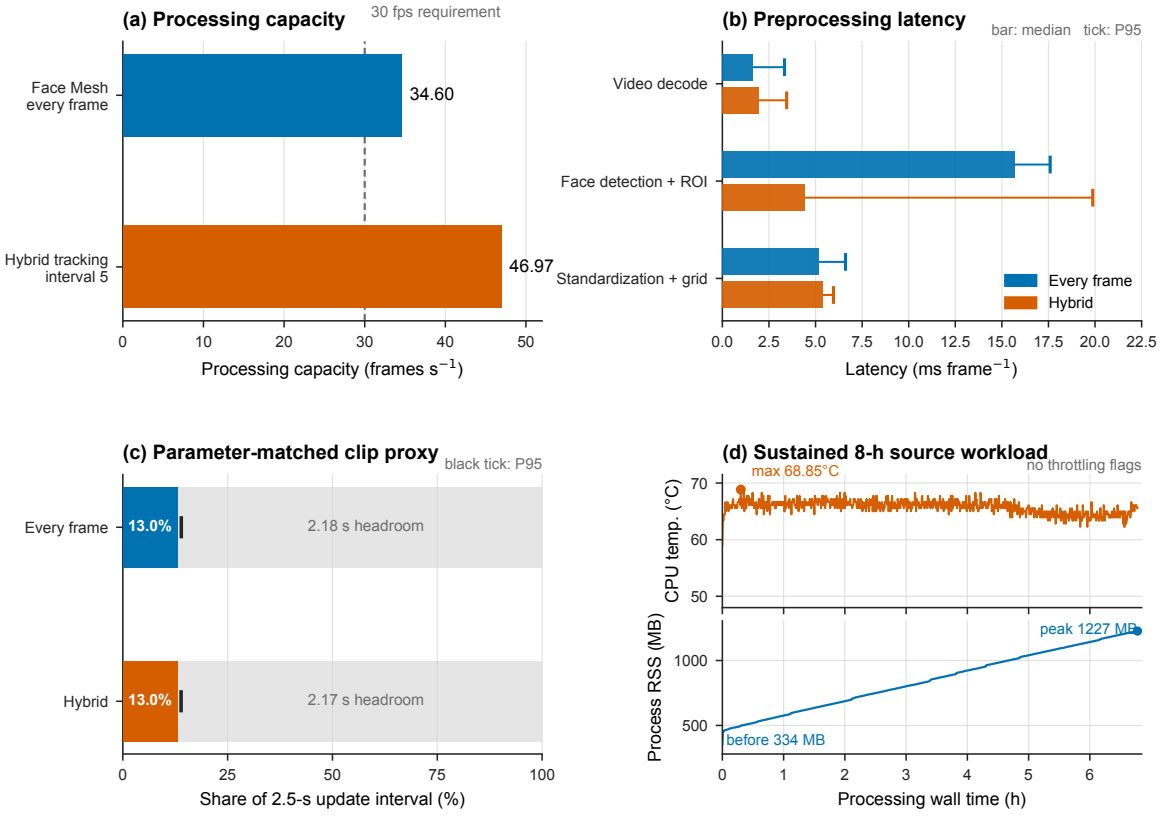


Fig. 4 Embedded real-time profiling on Raspberry Pi 5. (a) Processing capacity relative to the 30-fps requirement. (b) Per-frame preprocessing latency; bars are medians, terminal ticks P95. (c) Median share of the 2.5-s update interval taken by the 260,000-parameter inference scaffold; black ticks are P95, grey segments remaining headroom. (d) CPU temperature and profiler-process RSS on aligned time axes over 8 h of source-equivalent video in 6.78 h wall time, sampled every 10 s with the source looped continuously. The 605.38 MiB cleanup drop is close to the 596.61 MiB size of an $814{,}576 \times 192$ float32 feature array, which supports a profiling-buffer interpretation without establishing the bounded deployment working set. Panels (a) and (b) measure the streaming front end; panel (c) is a parameter-matched allowance, not trained SIR-PhysNet latency. No HR accuracy was computed on the device.

7 Conclusion

SIR-PhysNet is a camera-based HR pipeline organized around a 2.5-s update deadline. Signal-informed standardization absorbs the nuisance factors in front of the network, a network of 0.26 M parameters and 0.12 GFLOPs models the remaining pulse dynamics, and a parameter-free HR-DFF readout converts its tokens into a spectrally aligned surrogate. The method reaches an SD of 2.76 BPM, an RMSE of 2.75 BPM and $r = 0.98$ on UBFC-rPPG, and MAE and RMSE of 3.39 and 4.80 BPM on ECG-Fitness; since the ECG-Fitness baselines are literature-reported rather than protocol-matched, that result places the model in a comparable error range at a substantially smaller parameter count. On a Raspberry Pi 5 the streaming front end sustains 34.60 fps under per-frame face detection, the schedule under which the accuracy figures were obtained, and 46.97 fps under detect-and-track scheduling; it holds above 30 fps across 814,576 frames at a 68.85 °C peak with no throttling flag raised, and a parameter-matched scaffold at the same scale consumed roughly 13% of the stride interval. On this class of hardware the binding constraint is therefore face processing rather than network capacity or temperature.

Several limitations bound these claims and define the work that follows from them. Accuracy was measured only under intra-dataset subject-disjoint protocols, so transfer across sensors, frame rates, compression settings, demographics and uncontrolled illumination remains untested, and the last is a particular concern because illumination invariance is assigned here to a fixed preprocessing stage rather than to the network [29]. Reported values are means over three seeds whose individual runs were not retained, so no per-seed intervals are available. Component attribution is partial: the ablation removes the readout alongside the preprocessing stages but does not compare HR-DFF against a fixed physiological-band filter or a capacity-matched waveform decoder, and HR-DFF assumes a single dominant cardiac frequency per clip, which places beat-level HRV, arrhythmia and clinical claims outside the present scope. The deployment study covers one fan-cooled platform and one looped prerecorded video, the clip-level network cost is a parameter-matched allowance rather than trained-graph latency, the

detect-and-track schedule is characterized by cost and valid-face rate alone, and the instrumented RSS trace does not establish deployment-memory stability. Direct end-to-end timing of the trained graph, a memory-bounded rerun of the profiling harness, a calibrated power measurement, optional fixed-point optimization and a signal-fidelity study of ROI propagation under the tracked schedule are the measurements that would convert these results into a deployment claim.

Statements and Declarations

Funding This work was supported in part by the National Natural Science Foundation of China (No. 62427807).

Competing interests The authors declare that they have no competing financial or non-financial interests.

Ethics approval Approval from an institutional review board was neither sought nor required for this work, because no data were collected from human participants by the authors. All quantitative results were obtained from the publicly released UBFC-rPPG, MAHNOB-HCI and ECG-Fitness datasets, for which ethical approval and participant consent were obtained by the original dataset providers and which were used solely under their respective access conditions. The only recordings made by the authors are the illustrative frames in Fig. 1 and the prerecorded clip used as the profiling workload in Sect. 5; both are of one of the authors and were made by that author for the sole purpose of preparing this manuscript. No third party was recorded, and no private or unreleased human data of any kind were gathered for this study.

Consent for publication The facial frames reproduced in Fig. 1 are of one of the authors, who gives written consent to the publication of these images in print and in electronic form, including on the internet. No identifiable frame from UBFC-rPPG, MAHNOB-HCI or ECG-Fitness is reproduced in this manuscript, and no individual other than the author concerned appears in any figure.

Data availability UBFC-rPPG, MAHNOB-HCI and ECG-Fitness are available from their respective providers under the providers' access conditions. The aggregate Raspberry Pi profiling records including the per-run RSS accounting of Sect. 5.3, the plotting script for Fig. 4 and a profiling harness documenting the measurement procedure of Sect. 5.1 are included as supplementary material. The harness is a later reconstruction of that procedure rather than the program that produced the tabulated runs, so its output constitutes a new measurement. No participants were recruited for the embedded profiling study.

Author contributions H. Chai: conceptualization, methodology, software, experiments, analysis, visualization, and original draft. Z. Liu: conceptualization, literature review, visualization, data curation, and writing. W. Zhang: literature review, visualization, data curation, editing, and formatting. F. Jiang: supervision, funding acquisition, project administration, and review and editing. All authors reviewed and approved the manuscript.

References

1. Adjei, T., Knaack, L., Hertenstein, E., Wilson, H., Wilson, M., McNamara, S., Wilson, S., de Zambotti, M.: Pulse rate variability is not the same as heart rate variability: findings from a large, diverse clinical population study. Frontiers in Digital Health **7**, 1589498 (2025)
2. Ahmed, S.G., Verbert, K., Zaki, N., Aljassmi, H., Alnajjar, F., Siedahmed, N.: AI innovations in rppg systems for driver monitoring: Comprehensive systematic review and future prospects. IEEE Access **13**, 22893–22918 (2025)
3. Bobbia, S., Macwan, R., Benezeth, Y., Mansouri, A., Dubois, J.: Unsupervised skin tissue segmentation for remote photoplethysmography. Pattern Recognition Letters **124**, 82–88 (2019)
4. Bouraffa, T., Koutsakis, D., Zelvyte, S.: Deep learning-based rppg models towards automotive applications: A benchmark study. In: Proceedings of the Winter Conference on Applications of Computer Vision (WACV) Workshops, pp. 1171–1180 (2025)
5. Chen, S., Wong, K.L., Chin, J.W., Chan, T.T., So, R.H.: Diffphys: Enhancing signal-to-noise ratio in remote photoplethysmography signal using a diffusion model approach. Bioengineering **11**(8), 743 (2024)
6. Chen, W., McDuff, D.: Deepphys: Video-based physiological measurement using convolutional attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 349–365 (2018)
7. Chu, S., Xia, M., Yuan, M., Liu, X., Seppänen, T., Zhao, G., Shi, J.: Codephys: Robust video-based remote physiological measurement through latent codebook querying. IEEE Journal of Biomedical and Health Informatics (2025). DOI 10.1109/JBHI.2025.3540134. Early access
8. Contoli, C., Calisti, L., Di Fabrizio, G., Kania, N., Bogliolo, A., Lattanzi, E.: A power-aware vision-based virtual sensor for real-time edge computing. Journal of Real-Time Image Processing **21**, 103 (2024). DOI 10.1007/s11554-024-01482-0
9. Costa, C., Brito, J.H.: Throughput impact of software multithreading for deep-learning inference on the AMD Kria KV260. Journal of Real-Time Image Processing **23**, 89 (2026). DOI 10.1007/s11554-026-01889-x
10. Di Lernia, D., Finotti, G., Tsakiris, M., Riva, G., Naber, M.: Remote photoplethysmography (rPPG) in the wild: Remote heart rate imaging via online webcams. Behavior Research Methods (2024)
11. Finnegan, E., Davidson, S., Harford, M., Jorge, J., Watkinson, P., Young, D., Tarassenko, L., Villarroel, M.: Pulse arrival time as a surrogate of blood pressure. Scientific Reports **11**, 22767 (2021)
12. de Haan, G., Jeanne, V.: Robust pulse-rate from chrominance-based rppg. IEEE Transactions on Biomedical Engineering **60**(10), 2878–2886 (2013)
13. Hasan, Z., Ahmed, M., Faridee, A.Z.M., Purushotham, S., Kwon, H., Lee, H., Roy, N.: Demo: RhythmEdge: Enabling contactless heart rate estimation on the edge. In: IEEE International Conference on Smart Computing (SMARTCOMP) (2022)

14. He, Y., Zhao, J., Wang, L., Zhang, X.: Litesyncnet: A lightweight and robust rppg method with multi-scale feature fusion and frequency-domain complex loss for real-time heart rate estimation. Biomedical Signal Processing and Control **107**, 107758 (2025)
15. Kartynnik, Y., Ablavatski, A., Grishchenko, I., Grundmann, M.: Real-time facial surface geometry from monocular video on mobile gpus. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (2019)
16. Lee, E., Chen, E., Lee, C.Y.: Meta-rppg: Remote heart rate estimation using a transductive meta-learner. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 735–751. Springer (2020)
17. Lee, J.S., Hwang, G., Ryu, M., Lee, S.J.: LSTC-rPPG: Long short-term convolutional network for remote photoplethysmography. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 6015–6023 (2023)
18. Li, X., Chen, J., Zhao, G., Pietikainen, M.: Remote heart rate measurement from face videos under realistic situations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4264–4271 (2014)
19. Liu, X., Fromm, J., Patel, S., McDuff, D.: Multi-task temporal shift attention networks for on-device contactless vitals measurement. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 33, pp. 19400–19411 (2020)
20. Liu, X., Hill, B., Jiang, Z., Patel, S., McDuff, D.: Efficientphys: Enabling simple, fast and accurate camera-based cardiac measurement. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 5008–5017 (2023)
21. Liu, X., Narayanswamy, G., Paruchuri, A., Zhang, X., Tang, J., Zhang, Y., Sengupta, S., Patel, S., Wang, Y., McDuff, D.: rppg-toolbox: Deep remote ppg toolbox. arXiv preprint arXiv:2210.00716 (2023)
22. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M.G., Lee, J., Chang, W.T., Hua, W., Georg, M., Grundmann, M.: Mediapipe: A framework for building perception pipelines (2019). DOI 10.48550/arXiv.1906.08172
23. McDuff, D.J., Blackford, E.B., Estepp, J.R.: The impact of video compression on remote cardiac pulse measurement using imaging photoplethysmography. In: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), pp. 63–70. IEEE (2017)
24. Moço, A.V., Stuijk, S., de Haan, G.: Skin inhomogeneity as a source of error in remote PPG-imaging. Biomedical Optics Express **7**(11), 4718–4733 (2016)
25. Narayanswamy, G., Liu, Y., Yang, Y., Ma, C., Liu, X., McDuff, D., Patel, S.: Bigsmall: Efficient multi-task learning for disparate spatial and temporal physiological measurements. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 5008–5017 (2024)
26. Niu, X., Han, H., Shan, S., Chen, X.: Synrhythm: Learning a deep heart rate estimator from general to specific. In: 2018 24th International Conference on Pattern Recognition (ICPR), pp. 3580–3585. IEEE (2018)
27. Panousopoulos, V., Papaloukas, E., Leon, V., Soudris, D., Koumandakis, E., Lentaris, G.: HW/SW co-design on embedded SoC FPGA for star tracking optimization in space applications. Journal of Real-Time Image Processing **21**, 16 (2024). DOI 10.1007/s11554-023-01391-8
28. Poh, M.Z., McDuff, D.J., Picard, R.W.: Advancements in noncontact, multiparameter physiological measurements using a webcam. IEEE Transactions on Biomedical Engineering **58**(1), 7–11 (2011)
29. Shao, H., Luo, L., Qian, J., Yang, J.: Remote photoplethysmography in real-world and extreme lighting scenarios. arXiv preprint arXiv:2504.07924 (2025)
30. Soleymani, M., Lichtenauer, J., Pun, T., Pantic, M.: A multimodal database for affect recognition and implicit tagging. IEEE Transactions on Affective Computing **3**(1), 42–55 (2012)
31. Speth, J., Vance, N., Flynn, P., Bowyer, K., Czajka, A.: Unifying frame rate and temporal dilations for improved remote pulse detection. Computer Vision and Image Understanding **210**, 103246 (2021)
32. Spetlik, R., Cech, J., Matas, J.: Visual heart rate estimation with convolutional neural network. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
33. Sun, Z., Li, X.: Contrast-phys+: Unsupervised and weakly-supervised video-based remote physiological measurement via spatiotemporal contrast. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(5), 3639–3651 (2024)
34. Tulyakov, S., Alameda-Pineda, X., Ricci, E., Yin, L., Cohn, J.F., Sebe, N.: Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2396–2404 (2016)
35. Vance, N., Flynn, P.: Refining remote photoplethysmography architectures using CKA and empirical methods. arXiv preprint arXiv:2401.04801 (2024)
36. Verkruysse, W., Svaasand, L.O., Nelson, J.S.: Remote plethysmographic imaging using ambient light. Optics Express **16**(26), 21434–21445 (2008)
37. Wang, K., Tang, J., Wang, Y., Liu, X., Fan, Y., Ji, J., Shi, Y., McDuff, D.: Facephys: State of the heart learning. arXiv preprint arXiv:2512.06275 (2025)
38. Wang, W., den Brinker, A.C., Stuijk, S., de Haan, G.: Algorithmic principles of remote-ppg. IEEE Transactions on Biomedical Engineering **64**(7), 1479–1491 (2017)
39. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19 (2018)
40. Yang, Z., Wang, H., Lu, F.: Assessment of deep learning-based heart rate estimation using remote photoplethysmography under different illuminations. IEEE Transactions on Human-Machine Systems **52**(6), 1236–1246 (2022)
41. Yu, Z., Li, X., Niu, X., Shi, J., Zhao, G.: Autohr: A strong end-to-end baseline for remote heart rate measurement with neural searching. IEEE Signal Processing Letters **27**, 1245–1249 (2020). DOI 10.1109/LSP.2020.3007086
42. Yu, Z., Li, X., Niu, X., Shi, J., Zhao, G.: Physformer: Facial video-based physiological measurement with temporal difference transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(2), 2479–2495 (2023)
43. Yu, Z., Li, X., Zhao, G.: Remote photoplethysmograph signal measurement from facial videos using spatio-temporal networks. In: Proceedings of the British Machine Vision Conference (BMVC) (2019)
44. Yu, Z., Shen, Y., Shi, J., Zhao, H., Cui, Y., Zhang, J., Torr, P., Zhao, G.: PhysFormer++: Facial video-based physiological measurement with SlowFast temporal difference transformer. International Journal of Computer Vision **131**(6), 1307–1330 (2023)
45. Yu, Z., Shen, Y., Shi, J., Zhao, H., Torr, P.H., Zhao, G.: Physformer: Facial video-based physiological measurement with temporal difference transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4186–4196 (2021)
46. Zou, B., Guo, Z., Hu, X., Ma, H.: Rhythmmamba: Fast, lightweight, and accurate remote physiological measurement. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), vol. 39, pp. 11077–11085 (2025)