

Time-Series U-Net with Recurrence for Noise-Robust Imaging Photoplethysmography

Shenoy, Vineet; Wu, Shaoju; Comas, Armand; Lohit, Suhas; Mansour, Hassan; Marks, Tim K.

TR2025-145 October 17, 2025

Abstract

Remote estimation of vital signs enables health monitoring for situations in which contact-based devices are either not available, too intrusive, or too expensive. In this paper, we present a modular pipeline for pulse signal estimation from video of the face that achieves state-of-the-art results on publicly available datasets. Our imaging photoplethysmography (iPPG) system consists of three modules: face and landmark detection, time-series extraction, and pulse signal/pulse rate estimation. The pulse signal estimation module, which we call TURNIP (Time-Series U-Net with Recurrence for Noise-Robust Imaging Photoplethysmography), allows the system to faithfully reconstruct the underlying pulse signal waveform and uses it to measure pulse rate and pulse rate variability metrics, even in the presence of motion. When parts of the face are occluded due to extreme head poses, our system explicitly detects such “self-occluded” regions and maintains estimation robustness despite the missing information. Our algorithm provides reliable pulse rate estimates without the need for specialized sensors or contact with the skin, outperforming previous iPPG methods on both color (RGB) and near-infrared (NIR) datasets.

IEEE Access 2025

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 00000/ACCESS.000000

Time-Series U-Net with Recurrence for Noise-Robust Imaging Photoplethysmography

VINEET R. SHENOY¹, SHAOJU WU², ARMAND COMAS³, SUHAS LOHIT⁴, HASSAN MANSOUR⁴,
TIM K. MARKS⁴

¹Johns Hopkins University, Baltimore, MD 21210 USA (e-mail: vshenoy4@jhu.edu)

²Boston Children's Hospital, Boston, MA 02115 USA (e-mail: shaoju.wu@childrens.harvard.edu)

³Northeastern University, Boston, MA 02115 (email: comasmassague.a@northeastern.edu)

⁴Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139 (email: {slohit, mansour, tmarks}@merl.com)

Corresponding author: Suhas Lohit (e-mail: slohit@merl.com).

VS, SW, and AC performed this work as interns at MERL.

ABSTRACT Remote estimation of vital signs enables health monitoring for situations in which contact-based devices are either not available, too intrusive, or too expensive. In this paper, we present a modular pipeline for pulse signal estimation from video of the face that achieves state-of-the-art results on publicly available datasets. Our imaging photoplethysmography (iPPG) system consists of three modules: face and landmark detection, time-series extraction, and pulse signal/pulse rate estimation. The pulse signal estimation module, which we call TURNIP (Time-Series U-Net with Recurrence for Noise-Robust Imaging Photoplethysmography), allows the system to faithfully reconstruct the underlying pulse signal waveform and uses it to measure pulse rate and pulse rate variability metrics, even in the presence of motion. When parts of the face are occluded due to extreme head poses, our system explicitly detects such “self-occluded” regions and maintains estimation robustness despite the missing information. Our algorithm provides reliable pulse rate estimates without the need for specialized sensors or contact with the skin, outperforming previous iPPG methods on both color (RGB) and near-infrared (NIR) datasets.

INDEX TERMS Heart rate estimation, Signal Denoising, Vital Sign Estimation, RPPG, iPPG

I. INTRODUCTION

Health monitoring can improve the lives and increase the lifespans of all people, healthy or not. And with an increasing aging population globally, health monitoring can ease health difficulties for both the aging and their caregivers. From small commercial devices such as smart watches to large clinical machines such as CT scanners, medical devices keep physicians, patients, and everyday users aware of their health. These devices, however, are themselves a barrier to health monitoring—commercial devices tend to be expensive, while clinical devices are only accessible in medical facilities. Furthermore, medical equipment that requires physical contact with the body can be invasive and uncomfortable, limiting wider adoption of ubiquitous health monitoring technologies.

The COVID-19 pandemic has renewed interest in non-contact measurements of vital signs [1], [2], including pulse rate [3]–[8], breathing rate [9]–[12], blood pressure [13], and pulse transit time [9]. Remote healthcare has allowed patients to receive quality care even when offices are closed,

leading to healthier and safer lives. Beyond healthcare, this type of monitoring could potentially be used in safety-critical applications such as driver-monitoring [8], [14] and heavy equipment operation. Measuring quantities such as pulse rate and pulse rate variability—defined as fluctuations in the interval between successive beats of a heart, measured using PPG techniques—and doing so from facial video only can be used from hospital-based settings, to consumer electronics, and even safety-critical applications.

Imaging photoplethysmography (iPPG), also known as remote photoplethysmography (rPPG), is the process of determining the pulse rate and/or pulse waveform from non-contact video of the skin (e.g., the face). The key to accurately estimating the pulse waveform is to first measure the variation in the image intensity of the skin that contains the underlying pulse signal, which is weak and noisy, then extract the pulse signal through denoising [5], [15] or signal processing-based estimation [16]–[18]. Many pre-deep-learning methods [5], [15]–[17] first detect the face in each video frame, then

average the RGB pixel intensities across the entire face region in each frame, to obtain a 3-channel (R, G, B) time series. These algorithms estimate the underlying pulse signal from the spatially averaged 3-channel signal either by de-mixing under model-free assumptions of blind source separation [5], [15] or by projecting the RGB signals onto different color subspaces [16], [17]. Deep learning-based methods discussed in [3], [4], [19] input video frames directly and use attention modules to extract the pulse signal from the face region.

The SparsePPG [14] and AutoSparsePPG [8] methods first segment the face into regions and record the variation in the mean intensity value of each region over time. Unlike approaches presented in [5], [15]–[18], which extract a 3-channel (R, G, B) time series from the video, these methods [8], [14] extract a multi-channel time series whose channels represent facial regions rather than colors. The model estimates the pulse rate from the multi-channel time series by assuming that the pulse signal is sparse in the frequency domain and uses sparsity-promoting algorithms to find the underlying heart-rate frequencies that are shared across facial regions. Our proposed method adopts a similar multi-region analysis, but rather than using sparsity-driven algorithms as presented in [8], [14], we develop a network architecture that learns to extract the underlying pulse signal from the multi-region time series.

We adopt a modular framework for pulse signal estimation that achieves state-of-the-art results on publicly available datasets. We demonstrate the effectiveness of our algorithm in two imaging domains: the color (RGB) domain and the near-infrared (NIR) domain. Our model can recover the pulse rate in the presence of substantial motion, and our system's detection and special handling of *self-occluded* landmarks (facial landmarks that are occluded by other parts of the face) makes our method more robust to extreme head poses such as profile views. In addition, we explore a pulse rate variability analysis, which reflects neurocardiac function and autonomic nervous system activity. Our algorithmic paradigm includes three modules: a face and landmark detection module, a time-series extraction module, and a pulse signal estimation module called TURNIP (Time-Series U-Net with Recurrence for Noise-Robust Imaging Photoplethysmography¹). The system reconstructs the underlying pulse signal, which is then used for pulse rate and pulse rate variability estimation. The face detection and landmark detection module, as well as the TURNIP iPPG estimation module, contain deep-learning components, while the time-series extraction module explicitly records the temporal variation of the light intensity within each facial region. As a combined system, these three modules determine the pulse rate with state-of-the-art accuracy, even when head motion is present in the data.

¹An earlier version of this work appeared in [20], where the “N” in TURNIP stood for Near-Infrared. In the current paper, we expand the algorithm to operate on RGB input, enable the handling of occlusions, and employ an improved face and landmark detector. We also evaluate our algorithm on a larger dataset, demonstrate improved performance relative to state-of-the-art deep-learning methods, and add a pulse rate variability analysis to the paper.

To summarize, our contributions are as follows:

- We design a modular, interpretable pipeline for pulse-rate estimation and pulse rate variability from face videos; this pipeline consists of a face and landmark detection module, a time-series extraction module, and a pulse signal estimation module.
- We propose TURNIP, a time-series U-Net with Gated Recurrent Units (GRUs) as pass-through connections, which reconstructs the underlying pulse signal.
- For color (RGB) videos, we demonstrate that the ratio of the red and green color channels [17], [21], [22] in a spatio-temporal neural network better denoises input signals than raw color channels, leading to improved pulse signal estimation.
- We handle extreme head poses and poorly framed video by automatically identifying self-occluded or outside-of-frame landmarks, enabling our method to learn to handle bad information and be robust to extreme poses.
- We evaluate our algorithm on three publicly available datasets from the RGB and NIR domains, achieving state-of-the-art results on all datasets.

The rest of the paper is organized as follows: in Section II, we discuss related work. This is followed by our pulse signal and pulse rate estimation technique in Section III. Datasets used for evaluation are discussed in Section IV. The implementation details and experimental results are described in Section V, and we conclude in Section VI.

II. RELATED WORK

A. PULSE RATE ESTIMATION

Pulse Signal estimation can be separated into signal processing-based methods [5], [8], [14]–[17], [23] and deep neural network methods [3], [4], [7], [24]–[32]. We discuss each individually below.

1) Signal Processing-Based Methods

Signal processing-based methods include blind source separation (BSS) methods such as [5], [15] and model-based methods such as [14], [16], [17]. Blind Source Separation techniques [5], [15] consider the measured signal to contain both the underlying pulse signal and noise. To separate the signal from the noise, they use Principal Component Analysis (PCA) or Independent Component Analysis (ICA), depending on whether they desire the projected data to lie in the coordinate systems of maximum variance or statistical independence. Unlike these BSS methods, which do not consider skin-reflection models such as [18], CHROM [17] explicitly considers a subject's skin color as well as the light source upon the skin. It does so by eliminating the specular reflection component and white-balancing the underlying pulse signal using a standardized skin-tone vector. PBV [16] restricts all color variations to the pulsatile direction, assumes that the pulsatile signal is uncorrelated with other signal sources, and solves for a projection vector to obtain the pulsatile signal.

The methods SparsePPG [14] and AutoSparsePPG [8] are built upon sparse recovery algorithms. Recognizing that the

pulse-rate signal is quasi-periodic in time and sparse in the frequency domain, these methods seek to recover the peak frequency coefficients of the pulse signal. After segmenting the face into five regions and extracting a time-series from pixel intensities across video frames, these methods solve an optimization problem in which they seek to extract the underlying sparse set of frequency coefficients that describe the periodicity of the pulse signal. They assume that the set of active frequencies that correspond to the pulse signal should be the same across the five facial regions, and they use techniques in joint sparsity to solve for the target signal.

2) Deep-Learning Methods

Deep learning methods have recently surpassed classical signal processing-based methods in heart rate estimation performance. PhysNet [33] was among the first end-to-end methods to use spatio-temporal neural networks to reconstruct a pulse waveform directly from raw RGB video. Some methods, such as [3], obviate the need for explicitly-defined signal extraction techniques such as those described in the previous paragraph. Using the skin reflection model defined in [18], [3] inputs the difference between consecutive frames, rather than the frames themselves, to eliminate the stationary skin reflection color. The algorithm then uses the MSE loss between this difference signal and the corresponding difference signal of the ground-truth waveform. To account for motion, an attention mechanism is developed using soft-attention masks from 1×1 convolutions that are multiplied with the motion model feature map; the result only highlights the skin region for signal extraction. Follow-up work [4] improves on this attention mechanism by claiming that changes in background and “distraction” regions (such as the hair) can improve the quality of the attention mechanism to focus more on skin pixels. The work TS-CAN [19] added temporal shift modules [34] to [3] for better temporal processing, and MetaPhys [7] used this network in a meta-learning paradigm for few-shot adaptation across datasets in both supervised and unsupervised training procedures. A benefit of [3], [4], [7], [19], [24] is that these methods are end-to-end: RGB video frames are input, and the pulse signal is output. As a result, however, these methods become hard to interpret.

B. PULSE RATE VARIABILITY ESTIMATION

In addition to measuring the pulse rate, we explore our algorithm’s capability to measure pulse rate variability (PRV), which is closely related to heart rate variability (HRV). Pulse Rate Variability, defined by the time interval between adjacent heart beats and the fluctuations between them as measured through PPG, measures the heart-brain dynamic interactions and is closely associated with autonomic nervous system activity [35]. Fluctuations in the inter-beat interval (IBI) are due to the dynamic relationship between the sympathetic nervous system and the parasympathetic nervous system, as well as respiratory sinus arrhythmia and changes in tone of the vasculature [35]. Measuring these changes through pulse

rate variability can give insight into functioning of the heart, intestines, blood pressure, and more.

Measuring pulse rate variability, however, is highly dependent on the type of signal recorded (e.g. ECG vs PPG), the age of the subjects, and specifically, the length of the time recording. While 24-hour signal recordings are considered to be the “gold-standard” for measurement, shorter length signal may or may not correspond with the 24 hour measurements. To this end [36]–[40] have established correlations between shorter duration measurements and the gold-standard 24-hours recording. We follow the work of [38] detailing PRV for short term measurements using PPG waves, and report the power of the High Frequency (HF) components of the interbeat interval signal (in ms^2) and the root mean square of successive differences between normal heartbeats (RMSSD) measured in milliseconds (ms). For a full description of pulse rate variability, we refer the reader to [35].

III. METHOD

Our pulse rate estimation framework comprises three modules, outlined in black in Figure 1: a face and landmark detector, a time-series extractor, and a pulse signal estimator. Each component is described in the following subsections.

A. FACE AND LANDMARK DETECTION

Given the raw, unprocessed video, we detect the face box in each frame using [41], and the cropped faces are input to the LUVLi landmark detection algorithm [42]. In addition to estimating the locations of facial landmarks, LUVLi is unique in that it also estimates the uncertainty associated with each landmark’s location, as well as determining whether each landmark is self-occluded² (i.e., occluded by another part of the face).

In this work, we consider a landmark to be *invisible* if LUVLi determines either that it is self-occluded or that its location is outside of the image frame, as illustrated in Figure 3. Given that pulsatile signals in rPPG are weak, labeling landmarks as invisible is important to alert downstream modules of any noise that may corrupt the underlying PPG signal. We use these labels in generating the time series that are input to TURNIP, as described below. Previous methods such as [43] do not identify landmarks that are invisible, implicitly assuming that changes in face pose do not affect signal extraction. Other methods [3], [19], [24], [33] must learn to ignore unreliable facial regions using deep learning techniques such as attention.

B. TIME SERIES EXTRACTION

1) Generating Additional Landmarks

Given the landmark points in each frame and their visibility labels, we extract the temporal pulsatile signal, potentially noisy, at different regions in the face. To improve signal

²The LUVLi method [42] includes a *visibility estimator* that outputs a number in the range (0, 1) for each landmark, where outputs close to 0 (below a threshold) indicate that the landmark is self-occluded.

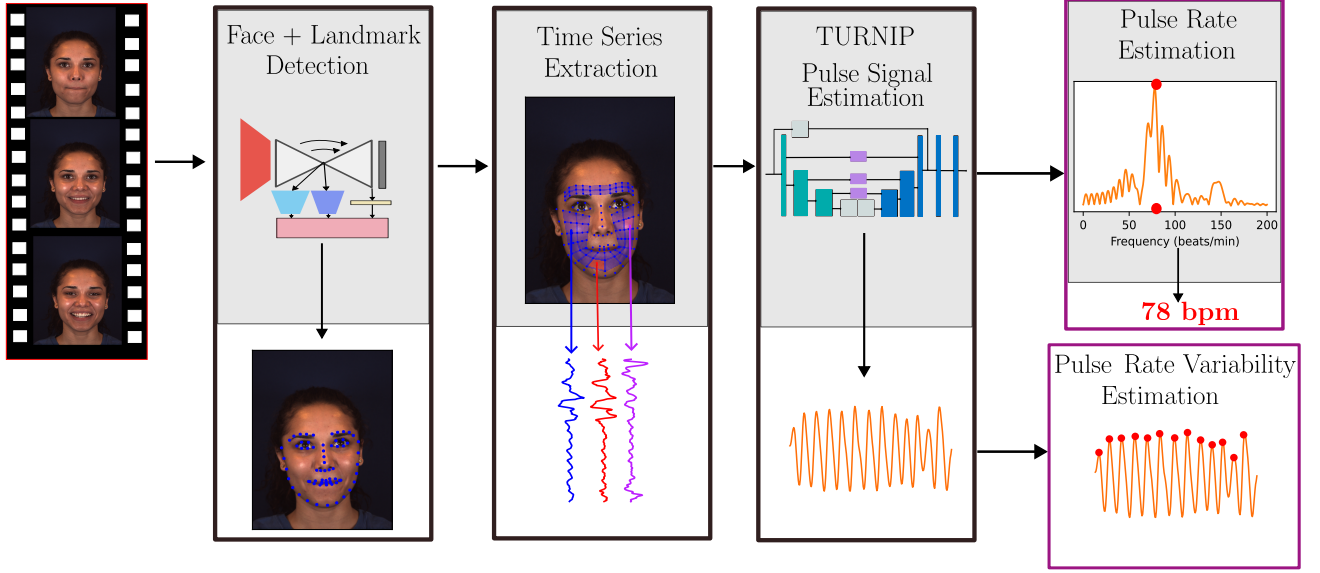


FIGURE 1: Our system for pulse signal estimation from video is composed of three modules, outlined in black: face and landmark detection, time series extraction, and pulse signal estimation. The pulse rate and pulse rate variability can then be estimated from the denoised pulse signal that is output from TURNIP.

reconstruction, we augment the number of landmarks by interpolating/extrapolating new landmarks on the cheeks, chin, and forehead as illustrated in Figure 2. We obtain the new landmarks on the cheeks and chin area by linearly interpolating between two existing landmarks. For example, we interpolate between the lower lip and the jawline to get chin landmarks. To extrapolate landmarks on the forehead, we first calculate a direction vector $\mathbf{v}_{\text{eyebrow}}$ along the right and left eyebrows by fitting a line to all visible eyebrow landmarks, then compute the perpendicular vector $\mathbf{v}_{\text{forehead}}$. We then extrapolate two rows of landmarks in the direction of $\mathbf{v}_{\text{forehead}}$, using one-fifth the distance between the inside eye corner and the corresponding mouth corner as the offset for each row. If one of the 68 landmarks is invisible (as defined in Section III-A), then we propagate its *invisible* label to every one of the augmented landmarks whose locations were determined using the invisible landmark. This is illustrated by the examples in Figure 3. Landmark augmentation from 68 to 145 landmarks helps to capture more regions of the face, including the critical forehead regions; this is particularly helpful in cases in which the pulsatile signal is weak or noisy in other regions. Using fewer regions of the face (e.g., using 5 large regions instead of 48 small regions) is also possible; we conduct an ablation study in Section V to explore this.

2) Handling Pixel Intensities

We incorporate three design features into the time series extraction module that contribute to our state-of-the-art results: **1)** We use these 145 landmark locations in each frame to define 48 facial regions, and within each region we average the pixel intensity values across all pixels in the region.

This step reduces noise. **2)** If a region is defined using an *invisible* landmark, we assign that region a large out-of-range value instead of averaging its intensity; this step recognizes that some signals will be inherently incorrect and noisy, and should be ignored or used for noise-robustness. **3)** For the RGB datasets (MMSE-HR and PURE), instead of extracting pixel intensities from a single color channel (e.g., the green channel of an RGB image), we extract a channel defined by the red-divided-by-green (R/G) intensity

$$\mathbf{x} = \frac{\mathbf{X}[:, 0]}{\mathbf{X}[:, 1]} \quad (1)$$

where $\mathbf{X}$ is the RGB pixel signal over T frames in dimension 0, and the color channels are in dimension 1; under the assumption that noise due to motion and illumination affects each color channel equally, the ratio of color channels should be robust to such noise as compared to individual color channels. All these steps differ significantly from previous work. Previous methods [30], [43] extract a signal by stacking all color channels from video in both the RGB and YUV space (and [30] uses [17], [18] as additional signals) and then hope that the neural network can extract the best information. Other methods [3], [4], [19], [44] input cropped video frames directly, and guide the neural network to extract the signal representation. Our method carefully designs the signal extraction technique, allowing the users to understand the inputs to the pulse signal estimation module.

3) Filtering and Normalizing the Signals

Finally, we temporally filter each of the 48 spatial regions using a fifth-order Butterworth filter with cutoff frequencies at 0.7 Hz and 4 Hz as in [4], corresponding to 42 beats per

minute (bpm) and 240 bpm, respectively. Next, we perform AC/DC normalization by subtracting the signal's mean, then dividing by the same mean:

$$\hat{\mathbf{y}}_i = \frac{(\mathbf{x}_i - \mu_i)}{\mu_i}, \quad (2)$$

$$\mathbf{y}_i = \frac{\hat{\mathbf{y}}_i}{\|\hat{\mathbf{y}}_i\|_2} \quad (3)$$

where $\mathbf{x}_i$ is the signal from region i , and μ_i is the temporal mean intensity in region i . The AC-DC normalized signal is $\hat{\mathbf{y}}_i$, after which we apply L_2 normalization in Equation (3); these two steps normalize our signal in the range $[-1, 1]$. The concatenation of $\mathbf{y}_i$ results in a 48-channel time series $\mathbf{y}_i \in \mathbb{R}^{48 \times 250}$ that is input to the TURNIP denoiser network, which estimates the pulse signal.

C. TURNIP PULSE SIGNAL ESTIMATION

The extracted time series is windowed into $T \times 48$ chunks for input into the TURNIP module, where T is the length of the time window. Segmenting the skin pixels into regions and spatially averaging them reduces noise; however, noise is still present due to facial deformations resulting from expressions, motion-related noise, and lighting variations, among other sources of noise. The network must learn to extract the pulse signal based on the statistics of the ground-truth data.

We design our architecture as a U-Net style neural network [45]. However, skip connections of the U-net in the form of gated recurrent units (GRUs) are used to incorporate temporal recurrence. The architecture of our pulse signal estimation module, which we call Time-Series U-net with Recurrence for Noise-Robust Imaging Photoplethysmography (TURNIP), is shown in Figure 4.

The $T \times 48$ signal is passed to the neural network as 48 channels. It goes through three stages of convolution (with kernel size 7) and downsampling, first by a factor of three and then by a factor of two. After reaching the lowest resolution, the signal is convolved and upsampled back to the original spatial and channel dimension over multiple stages. At every resolution of the U-net, we connect the encoding and decoding sub-networks by a skip connection module (each indicated by a purple rectangle in Figure 4). In parallel with each 1×1 convolutional skip connection (which is present in a typical U-net), we introduce a novel recurrent skip connection that uses gated recurrent units (GRUs) to provide temporally recurrent features. The output (hidden states) of this GRU layer is concatenated with the output of the standard (1×1) skip connection layer before being concatenated with the input to the corresponding convolution+upsampling layer. At each time scale, the convolutional layers of the U-net process all of the samples from the time window in parallel. In contrast, the new recurrent GRU layers process the temporal samples sequentially. This recurrence effectively extends the temporal receptive field at each layer of the U-net. After the last upsampling layer, we append a 1×1 convolutional layer to collapse the channel dimension to a single channel for appropriate calculation of a loss.

TABLE 1: Testing on three datasets in both the RGB and NIR domains

	MMSE-HR [46]	MR-NIRP Car [8]	PURE [47]
Domain	RGB	NIR	RGB
Resolution	1040 × 1392	640 × 640	640 × 480
Frame Rate (fps)	25	30	30
Codec	h264	h264	h264
No. Videos	102	19	60
No. subject	40	18	10
Male/Female Subjects	17/23	16/2	8/2
GT Signal Type	BP Wave	Pulse Ox	Pulse Ox
GT Sampling Rate (Hz)	1000	60	60

Training the TURNIP pulse signal estimation module requires an appropriate selection of loss function. We chose to minimize one minus the Pearson Correlation Coefficient, a measure of covariance between two variables. Consider two vectors $\bar{\mathbf{z}}$ and $\mathbf{z}_{\text{gt}}$ representing the predicted waveform and the ground-truth waveform of length T , respectively. We define a function $F(\bar{\mathbf{z}}, \mathbf{z}_{\text{gt}})$ such that

$$F(\bar{\mathbf{z}}, \mathbf{z}_{\text{gt}}) = 1 - \frac{T \cdot \bar{\mathbf{z}}^T \mathbf{z}_{\text{gt}} - \mu_{\bar{\mathbf{z}}} \mu_{\mathbf{z}_{\text{gt}}}}{\sqrt{(T \cdot \bar{\mathbf{z}}^T \bar{\mathbf{z}} - \mu_{\bar{\mathbf{z}}}^2)(T \cdot \mathbf{z}_{\text{gt}}^T \mathbf{z}_{\text{gt}} - \mu_{\mathbf{z}_{\text{gt}}}^2)}} \quad (4)$$

We then seek to minimize this function with respect to the parameters of the neural network

$$\theta^* = \arg \min_{\theta} F(\bar{\mathbf{z}}, \mathbf{z}_{\text{gt}}) \quad (5)$$

During training, we also design a novel data augmentation scheme that aims to capture lower and higher frequencies of the target pulse rate range, for which sufficient training data may not be available. In our "SpeedUp" augmentation, we crop the input window of length T by a random percentage between 20% and 40%, and linearly interpolate the samples back to the original window size T . For the "SlowDown" augmentation, we randomly chose a length that is 20% to 40% larger than our target time windows (e.g. $1.2 \times T$), extract the signal from the video, and linearly interpolate the signal to the target length T . In these ways our "SpeedUp" and "SlowDown" augmentation have extrapolated the statistics of the data to higher and lower frequencies respectively.

As compared to methods such as [30], [43] that ingest a variety of signal sources and transform them into signal output, or methods such as [3], [4], [24] that need to extract and estimate a signal from video, our method serves only as a denoiser. Our method takes a noisy signal estimate and outputs a cleaner signal estimate. In Section V, we show that our formulation produces state-of-the-art results on major datasets, and we conduct ablation studies to demonstrate the effectiveness of our design choices.

IV. DATASETS

We test our algorithm on three video datasets in both the RGB and Near-Infrared (NIR) modalities, and describe the datasets in Table 1 and below, as well as the challenges associated with each dataset. We train and test on each dataset independently, except for the cross-dataset evaluation described in

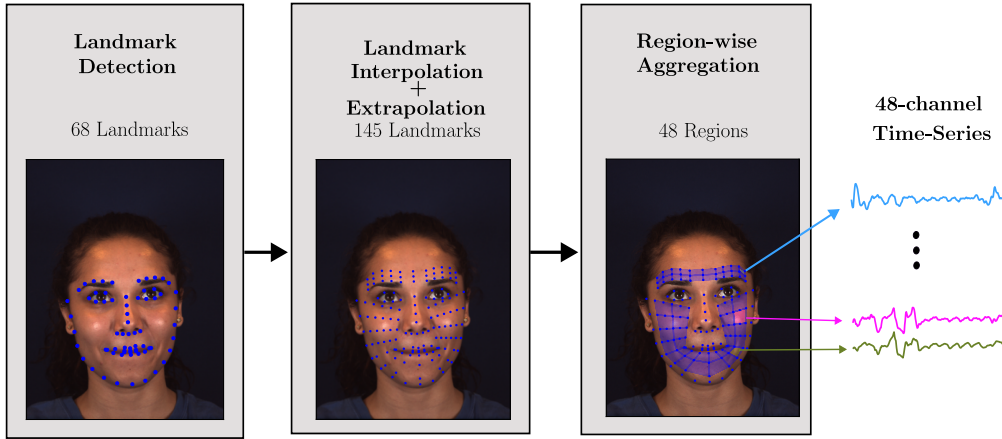


FIGURE 2: Generating of landmark and feature regions. We first start by detecting 68 landmarks from the LUVLi [42] landmark detector. We then interpolate these landmarks across the cheeks and chin and extrapolate them up the forehead, to generate 145 landmarks. We use these landmarks to define 48 regions. Finally, we aggregate the pixel intensities in each region using spatial averaging to obtain a 48-channel time-series.

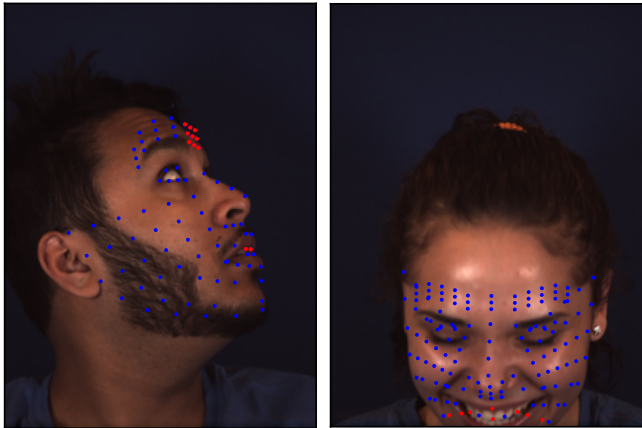


FIGURE 3: These example frames show the augmented set of 145 landmarks, with *invisible* landmarks shown in red. Landmarks are labeled *invisible* if they are either self-occluded (occluded by another part of the face, as illustrated in the left image) or outside of the image boundary (as illustrated in the right image). Finally, landmarks are also labeled invisible if their locations were determined by interpolating/extrapolating from an invisible landmark. Previous algorithms such as [8], [14] do not detect when landmarks are invisible, which means that such landmarks can cause previous methods to have incorrect results in frames that have extreme head rotations or translations. Because we explicitly detect invisible landmarks and label them as such, our algorithm can learn to be more robust to extreme poses.

Section V-D1. For each dataset, we downsample the ground-truth signal to the frame rate of the video. We then apply the AC-DC normalization, L_2 normalization, and bandpass filtering described in Section III-B3. We follow the evaluation

protocol of previous literature to ensure no data leakage.

MMSE-HR [46]: In the MMSE-HR dataset, various emotions are elicited from subjects while ground-truth blood-pressure waveforms (synchronized with the video) are captured using a finger sensor that was calibrated with a blood pressure cuff. This dataset contains considerable head motion and occlusions, to which our algorithms are robust. During training, we used 10-second windows ($T = 250$ samples), and shift the time window by 60 samples. Since we are performing leave-one-subject-out cross validation, this results in an average of 8026 windows for training and 9.7 windows for testing. During test time, we evaluate our model on 10-second samples with no overlap. We concatenate three 10-second windows and perform evaluation on 30-second windows to match the 30-second window evaluation protocol of [4].

MERL-Rice Near-Infrared Pulse (MR-NIRP) Car [8]: Recorded using an NIR camera with a 940 ± 5 nm band-pass filter, the MR-NIRP Car dataset includes facial videos (duration 2–5 min.) with synchronized ground-truth PPG waveforms collected using a fingertip pulse oximeter. The dataset is split into a “Driving” subset and “Garage” subset. During “Driving”, data were captured while driving through a city, resulting in illumination changes and head motion. There were 14 daytime videos and 4 videos that were captured at night. The “Garage” subset was recorded while the car was parked inside a garage, with less head motion and much less variation in illumination. As in [8], we evaluate only on the “minimal head motion condition” for all scenarios—we note, however, that even this scenario contains significant head motion in the Driving subset. An advantage of the NIR modality is that it minimizes illumination variations [8]. However, NIR frequencies introduce new challenges for iPPG, including weaker blood-flow-related intensity changes in the NIR portion of the spectrum and low signal-to-noise ratio

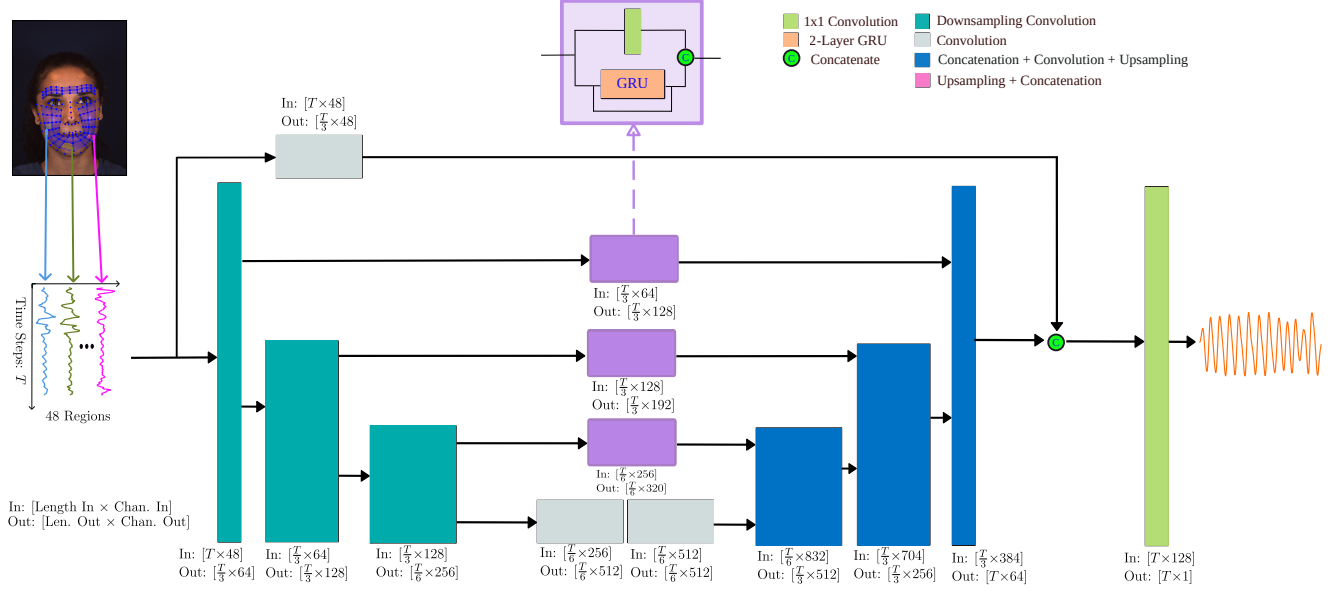


FIGURE 4: The TURNIP Pulse Signal Estimation module. The signals from the 48 individual regions are extracted at input to the network as a $T \times 48$ matrix. The spatio-temporal network denoises the signal to the statistics of the data, and outputs a clean signal. See Figure 6 for example input and output of TURNIP.

(SNR) due to reduced sensitivity of camera sensors. The results demonstrate that despite these challenges, our method is able to accurately estimate subjects' pulse rates.

PURE [47]: In the PURE dataset, subjects perform various head motion tasks while synchronized video and pulse waveform data (from a fingertip pulse oximeter) are captured. There are six head-motion tasks: *Steady*, *Talking*, *Slow Translation*, *Fast Translation*, *Small Rotation*, and *Medium Rotation*. We split the dataset into train/validation/test/splits as in [48], resulting in 7613 training windows and 136 test windows.

V. EXPERIMENTS AND RESULTS

A. EVALUATION PROTOCOL

1) Pulse Rate Estimation

To compute the pulse rate estimate, we first multiply the time-series by a hanning window to avoid spectral leakage. We then take the L -point FFT, where $L = 100 \times \text{signal length}$. We square the magnitude of the coefficients to get the power, and take the bin with highest power (among the positive frequencies) as our estimate of the pulse rate.

We follow the evaluation protocols as described in [4] and report the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) between the predicted pulse rate and the ground-truth pulse rate. The MAE is defined as

$$\frac{1}{N} \sum_{i=1}^N |R_i - \hat{R}_i| \quad (6)$$

and the RMSE is defined as

$$\sqrt{\frac{1}{N} \sum_{i=1}^N (R_i - \hat{R}_i)^2} \quad (7)$$

where R_i is the ground-truth pulse rate in time window i , $\hat{R}_i$ is the predicted pulse rate in time window i , and N is the total number of time windows. In addition, we report the PTE6 (Percent of Time that Error < 6 bpm), which is defined as

$$\text{PTE6} = \frac{100}{N} \sum_{i=1}^N P_i, \text{ where } P_i = \begin{cases} 1, & \text{if } |R_i - \hat{R}_i| < 6 \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

This quantity describes the percentage of pulse rate estimates that are within 6 beats per minute (bpm) of the ground-truth pulse rate. We chose this metric as it roughly encodes the notion of what percent of the time the estimated pulse rate is correct.

For a fair comparison against previous methods, we evaluate on 30-second time windows for the MMSE-HR dataset in accordance with evaluation protocols from [4], and on 10-second time windows for the MR-NIRP Car dataset to conform to our evaluation protocols in [20]. We evaluate on 30-second windows on the PURE dataset to conform with previous literature [48].

2) Pulse Rate Variability Metrics

We report the metrics as described in [35], [38] which enumerate the ultra-short duration metrics that correlate well with the 24-hours recordings. Given that signals in the PURE dataset are approximately 1 min long, we report the power of the High Frequency (HF) components of the interbeat interval in milliseconds squared (ms^2) and the root mean square of successive differences between normal heart beats (RMSSD) in milliseconds (ms) [35]. We report these numbers using the HeartPy [49], [50] python library, which standardizes

the computation of the above metrics. We do not report the low frequency components of the interbeat interval signal as this metrics needs at least 2 minutes of recording, nor the low-frequency/high-frequency ratio as this metric is most accurately reported on signals of 24-hr duration.

B. IMPLEMENTATION DETAILS

For face detection, as described in Section III-A, we use the off-the-shelf detector as trained by [41]. The time series extraction was described in detail in Section III-B. Below, we describe the training procedure for the landmark detection introduced in Section III-A for the MMSE-HR results and the TURNIP iPPG estimator described in Section III-C.

LUVLi Landmark Localization: In addition to accurate locations of facial landmarks, the LUVLi landmark detector [42] outputs a visibility for each landmark that indicates whether the landmark is “self-occluded” (signifying that the landmark is occluded by another part of the face, e.g., in the case of a profile face). If LUVLi determines that a landmark is invisible (self-occluded or outside of the image boundaries), then for every face region that is defined using that landmark’s location, our time series extraction module sets the region’s intensity value to -10, which is a large negative value outside of the normalized range of the signal. This enables TURNIP to learn to ignore those regions when estimating the pulse signal. We show that this novel use of landmark visibility serves as a pseudo-attention mechanism that improves pulse signal estimation.

TURNIP iPPG Estimation: We use the architecture shown in Figure 4. It consists of a three stage U-Net with 48 input channels; the time-resolution decreases while the channel dimension doubles at each stage until we reach 512 channels; we then decode this input by increasing the time-resolution and decreasing the channel resolution, adding GRU output along the way. The hidden state of the GRU is re-initialized for each time window of length T that is fed into the network. For the MMSE-HR dataset, we use the Adam optimizer with an initial learning rate of 1.5×10^{-3} and weight decay of 1×10^{-4} trained for 8 epochs. We found the same hyperparameters worked well for PURE. On the MR-NIRP Car dataset, we use a learning rate of 1.5×10^{-4} reduced at each epoch by a factor of 0.05. The learning rate is decayed by a factor of 0.99 at each epoch, and we train for 8 epochs. We use 10-second time window, and shift the window by 60 samples to generate our training set. Given the limited data, we use leave-one-subject-out cross validation for MMSE-HR and MR-NIRP Car datasets, and the train/val/test splits of the PURE dataset.

During test time, to replicate the 30-second-window evaluation process as described by [4] on the MMSE-HR dataset, we concatenate three 10-second output windows of TURNIP to form a 30-second evaluation window. On the MR-NIRP Car dataset, we follow the evaluation protocol we described in [20], which uses the same metrics as in [4] but evaluates on 10-second windows. To be consistent with [20] on the MR-NIRP Car dataset, we use the OpenFace landmark de-

TABLE 2: Results on the MMSE-HR dataset using 30-second windows (TURNIP results show mean and standard deviation across four random network initializations). The results for ICA [6], CHROM [17], and POS [18] are copied from [4]. AutoSparsePPG uses our signal extraction techniques, while [3], [4] have no analogous signal extraction technique.

Method	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
ICA [6]	5.44	12.00	-
CHROM [17]	3.74	8.11	-
POS [18]	3.90	9.61	-
AutoSparsePPG [8]	4.55	14.42	88.10
CAN [3]	4.06	9.51	-
InverseCAN [4]	2.27	4.90	-
Federated [53]	2.99	-	-
Physformer [54]	2.84	5.36	-
EfficientPhys-C [55]	2.91	5.43	-
ND-DeepPPG [56]	1.84	4.83	-
ContrastPhys [44]	1.11	3.83	-
TURNIP	1.04	2.84	93.02

TABLE 3: Results on the PURE dataset. The results listed are based on implementations of [57]

Method	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
CHROM [17]	2.07	9.92	-
POS [18]	5.44	12.00	-
HR-CNN [48]	1.84	2.37	-
CVD [43]	1.29	2.01	-
Gideon [58]	2.1	2.6	-
DualGAN [57]	0.82	1.31	-
Yue et. al [59]	1.23	2.01	-
ContrastPhys [44]	0.48	0.98	-
TURNIP	0.36	0.67	100

tector [51], [52], and smooth the resulting landmark locations using a 10-frame moving average. In the next subsection, we show that our model achieves state-of-the-art performance on the datasets.

C. RESULTS

1) Pulse Rate Estimation Analysis

We show our results on the MMSE-HR dataset in Table 2. On this challenging dataset, we reduce the RMSE error from 3.83 [44] to 3.46 bpm, and are second best in terms of MAE. Furthermore, compared to the previous deep-learning-based methods [3], [4], our modular system is more interpretable, as it does not involve the black-box attention mechanisms and signal reconstruction of these end-to-end approaches. Our pipeline has interpretable inputs and outputs that show exactly how a signal is extracted and how the underlying pulse waveform is estimated, which is a significant advantage over purely end-to-end methods. We see similar improvements on the PURE dataset, as shown in Table 3. We exceed performance on both signal processing-based methods as well as deep-learning methods. In addition, we perform a cross-dataset performance study, which we defer to the ablation studies.

In addition, we report our results using 10-sec time windows on the MMSE-HR dataset in Table 4, corresponding to

TABLE 4: Comparison of on the MMSE-HR dataset using 10-second vs. 30-second windows

Method	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
TURNIP (30-sec)	1.17±0.11	3.46±0.21	93.21±1.14
TURNIP (10-sec)	2.81±0.08	9.59±0.26	89.31±0.58

TABLE 5: Results on the MR-NIRP Car dataset

Method	MR-NIRP Car			
	Driving		Garage	
	RMSE (bpm) ↓	PTE6 (%) ↑	RMSE (bpm) ↓	PTE6 (%) ↑
DistancePPG [60]	>15	24.6	>15	37.4
SparsePPG [14]	>15	17.4	>15	35.6
AutoSparsePPG [8]	11.6	61.0	5.1	81.9
PhysNet-STSL-NIR [4]	13.2	53.2	6.3	88.8
TURNIP	11.4	65.1	4.6	89.7

the evaluation protocol we use on the MR-NIRP Car dataset. We believe that 10-sec windows are more appropriate for evaluation for several reasons: 1) Because 30-sec windows average the pulse rate over a long duration, short-term errors in pulse rate can be averaged out, making it seem as if a method estimates pulse rate more accurately than it actually does. (In fact, some of the videos in the MMSE-HR dataset are not much longer than 30 sec.) 2) In many real-world applications, shorter wait times are more desirable or necessary. 3) The (more challenging) 10-sec window scenario more closely resembles a real-time, instantaneous measurement of pulse rate, which may be important for clinical acceptance in the future. The results in Table 4 show that for 10-sec windows, the MAE and RMSE are higher and the PTE6 is lower; this demonstrates that the 10-sec evaluation protocol is more challenging, with more room for performance gains that may lead to further improvements in algorithms.

Our algorithm outperforms previous methods on the near-infrared MR-NIRP Car dataset [8] as well, as shown in Table 5. Note that for this dataset, we use the OpenFace landmark detector [52] rather than LUVLi [42], and we average the landmark locations across 10 frames as described in [20]. On both the Driving (city driving) subset and the Garage (car running while parked in a garage) subset of the MR-NIRP Car dataset, we achieve significantly higher PTE6 than previous methods, indicating that our algorithm captures the true pulse rate (within 6 bpm) a greater percentage of the time than previous methods. We also achieve smaller root-mean-squared error (RMSE) than previous methods on both subsets, showing that our method also reduces the error on a window-by-window basis. Even though the pulsatile signal is weaker in the NIR domain than in RGB, our method is still able to estimate the underlying pulse wave for pulse rate estimation more accurately than previous methods.

2) Statistical Analysis of Pulse Rate Estimation

We perform a modified Bland-Altman analysis—plotting the ground-truth pulse rate against the difference between the ground-truth and predicted pulse rate—for both the MMSE-

HR dataset and PURE dataset in Figure 5. On the MMSE-HR dataset, we see that we have a mean difference between predicted and ground-truth pulse rates of 0.48, which shows that our predictions accurately match the ground-truth measurements. Furthermore, only two of our estimates fall outside of the 95% limits of agreement, defined as $1.96 \times$ the standard deviation of the differences. The greatest difference between the predicted and ground-truth pulse rates occur at higher pulse rate ranges, which most likely means that there was insufficient training data at those rates.

We do the same for the PURE dataset and show the results in Figure 5 on 10-second windows to show a variety of heart-rate estimates. In both cases, we see a mean difference very close to zero. Also, nearly all of our pulse rate estimates fall within our limits of agreement, and the ones that are outside our limits are still within 2 bpm of the ground truth.

3) Pulse Rate Variability Analysis

We perform a HRV analysis on the predicted signals from TURNIP and the ground-truth and display the results in Figure 8. We do this on the PURE dataset, for which videos are at least 1-minute long. We note that our mean difference for the higher frequency component is -805.49 ms^2 while for the RMSSD metric our mean difference is -34.83 ms . For the high-frequency power estimate, many of our estimates have a difference close to zero, validating our ability to correctly predict the high frequency power. We notice a similar trend for the RMSSD metric. We believe that further research should focus on reconstructing waveform characteristics more effectively.

4) Qualitative Analysis

In Figure 6, each part ((a), (b), or (c)) shows example result waveforms for a single 10-sec time window from the test set of the MMSE-HR dataset. For each time window, the left column shows time-domain waveforms, and the right column shows the same signal in the frequency domain. In both the top and bottom rows, the signals in orange show the ground-truth pulse signal, while the overlaid signals in blue show the estimated signals. In the top row, the blue signal shows one channel (from one face region) of the output of our time series extraction module, *before* TURNIP pulse signal estimation. In the bottom row, the blue signal shows our system's final estimate of the pulse signal, *after* TURNIP pulse signal estimation. The peak frequency of each frequency domain graph provides the system's estimate of the pulse rate.

From the frequency domain graphs (lower right of each part), we can see that our TURNIP pulse signal estimator closely reconstructs the underlying spectrum, attenuating spurious peaks and generating an accurate pulse rate estimate. In Figure 6 (a), we see that the extracted time series signal (top row blue curves) has significant power at a lower frequency; our TURNIP algorithm attenuates this frequency and boosts the correct one. In (b), we see that the extracted time series signal is noisy in the frequency domain (blue curve at top right), with strong peaks at lower and higher frequencies

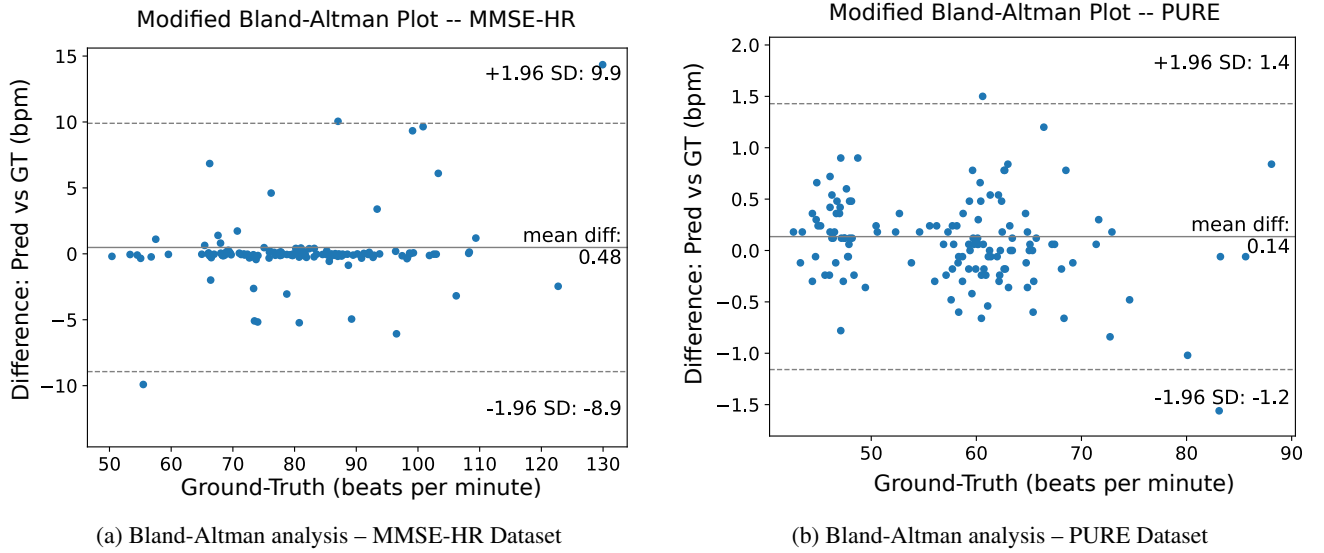


FIGURE 5: Bland-Altman Analysis. Each point in the MMSE-HR and PURE graphs represent a non-overlapping 10-second window.

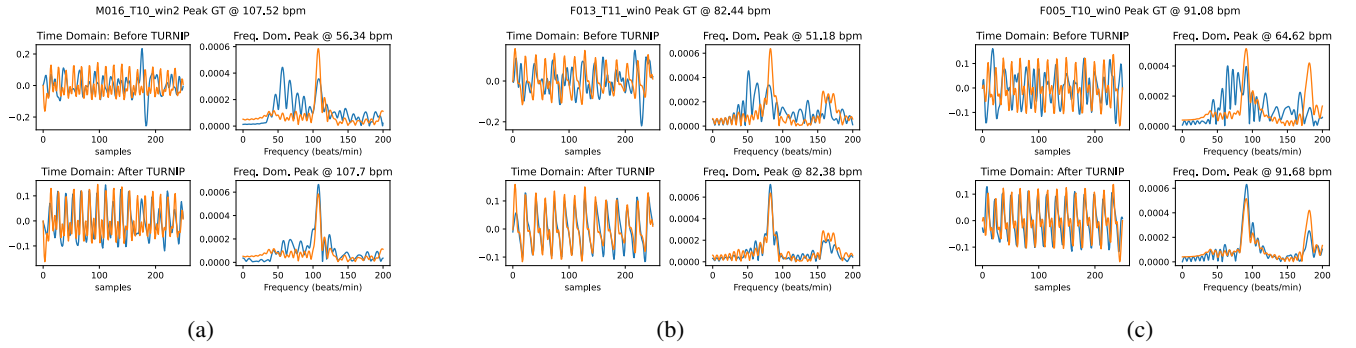


FIGURE 6: iPPG estimation for three different signals. The signals in orange are the ground-truth, with a peak frequency at 107.52 bpm in (a), 82.44 bpm in (b), and 91.08 bpm in (c), while the signals in blue are the inputs/outputs of our TURNIP pulse signal estimation algorithm. The top and bottom rows respectively show the estimate *before* and *after* TURNIP pulse signal estimation. Each row shows the time-domain signals on the left and frequency-domain power spectra on the right. The title of each frequency spectrum plot shows the peak frequency of the estimated signal.

TABLE 6: Cross Dataset Comparison

Train Dataset	Test Dataset	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
PURE	MMSE-HR	4.35	13.54	89.14
MMSE-HR	PURE	0.68	1.06	99.26

than the true pulse rate, and the predicted pulse rate before TURNIP is greater than 6-beats away from the ground-truth. Our TURNIP pulse signal estimator attenuates the spurious peaks and correctly predicts the true pulse rate (blue curve in bottom right). Figure 6 (c) shows similar behavior.

D. ABLATION STUDIES

1) Cross-Dataset Evaluation

In this experiment, we train on one dataset and test on another dataset without any further fine-tuning; we show the results

in Table 6. We see that MMSE-HR transfers very well to the PURE dataset. We believe this occurs because of the nature of the MMSE-HR dataset—firstly, there is more data upon which to train and secondly, it contains many examples of unconstrained motion with corresponding ground-truth signal from which the network can learn. The PURE dataset does not have the same distribution of unconstrained noise data. In addition to the significantly smaller dataset size, it does not contain the same noise characteristics that are typically found in the MMSE-HR dataset. Overall though, we still see good performance. When training on PURE and testing on MMSE-HR, we perform a Bland-Altman analysis and display the results in Figure 7. We see that, on average, we overestimate the ground-truth signal by approximately 3 beats per minute, but the majority of our performance degradation results from a few outliers. Therefore, we see that the majority

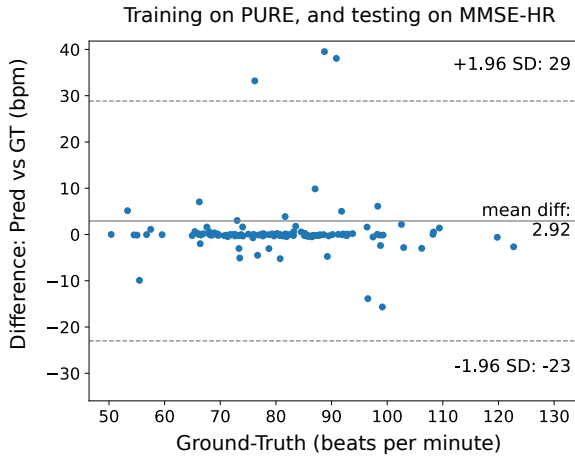


FIGURE 7: A modified Bland-Altman Analysis when training on PURE and testing on MMSE-HR. Each point represents a non-overlapping 10-second window of the MMSE-HR dataset. We see that we do well, except for a few large outliers.

TABLE 7: Pulse Rate estimation using different color channels and motion conditions

Dataset	Motion – Channel	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
MMSE-HR	Low Motion – Green	1.34	3.41	92.40
	Low Motion – RoG	0.80	2.33	96.20
	High Motion – Green	2.92	11.72	90.44
	High Motion – RoG	1.92	5.02	92.50
PURE	Low Motion – Green	2.17	7.78	93.95
	Low Motion – RoG	1.11	4.85	97.17
	High Motion – Green	2.23	7.27	94.33
	High Motion – RoG	1.99	7.18	95.28

of estimates, when training on PURE and testing on MMSE-HR, are within 5 beats of the true pulse rate.

2) Effect of Channel Ratio and Performance in low and high-motion conditions

In Table 7, we show the performance of TURNIP when the input channel is green or RedoverGreen (RoG) for low motion videos vs high motion videos. In the PURE dataset, we define as “high motion” the videos labeled by the dataset authors as *Slow Translation*, *Fast Translation*, and *Medium Rotation*, while the “low motion” videos are those labeled as *Steady*, *Talking*, and *Small Rotation*.

For each video in the MMSE-HR dataset, we measure the amount of motion in the video as follows. First, we compute the standard deviation across all frames of the 2D location of each of the 68 facial landmarks found by LUVLi [42]. Next, we compute the mean of these 68 standard deviations to obtain a single scalar measure of motion in the video. For each subject, the one video with the most motion is considered “high motion,” and the remaining videos of that subject are considered to be “low motion.” See the appendix for the exact splits. As we can see from the table, in all scenarios the RoG channel improves heart-rate estimation performance, whether

it be high motion or low motion. This shows the effectiveness of our use of color channels.

TABLE 8: Color Channel performance on the MMSE-HR dataset

Method	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
Red	3.66±0.87	9.22±1.32	82.94±0.44
Blue	44.93±2.31	47.33±4.01	1.55±0.52
Green	1.63±0.21	5.93±1.11	90.88±0.84
{Red, Green}	1.81±0.31	5.47±1.31	87.72±0.48
{Green, Blue}	2.45±0.61	8.03±1.05	89.14±0.33
{Red, Blue}	3.04±0.55	8.45±1.75	85.27±0.61
Red-over-Green (R/G)	1.17±0.11	3.46±0.21	93.21±1.14
Green-over-Red	3.76±0.21	8.56±0.44	77.51±0.51
Blue-over-Green	4.41±0.31	11.01±0.62	79.84±0.48
Green-over-Blue	4.67±0.18	11.09±0.18	78.29±0.18
Red-over-Blue	4.66±0.48	10.84±0.28	81.39±0.36
Blue-over-Red	4.72±0.61	10.76±0.32	78.29±0.22
{Red, Green, Blue}	2.93±0.18	10.13±0.47	89.14±0.15

3) Using multiple color channels

In Table 8, we experiment with different color channels as input to TURNIP, stacking the color channels into a matrix of size $T \times 48 \times 3$ for {Red,Green,Blue} input or $T \times 48 \times 2$ for 2-color input such as {Red,Green}. When we include the other color channels, including the weaker blue signal, we introduce in-domain noise that makes optimization more difficult. The 3-channel R,G,B TURNIP cannot replicate the more-optimal ratio of the red and green channels, nor does it learn how to handle the noise as effectively. We notice the best results when inputting the red over green channel directly, and report these results. This shows a simple modification to the signal extraction procedure in the RGB image domain can result in state-of-the-art performance.

4) Handling self-occluded landmarks

One of our contributions involves the detection and handling of self-occluded and out-of-frame landmarks during training, which improves model robustness during inference. We show in Table 9 the effects of incorporating or omitting the occlusion handling module. On the challenging MMSE-HR dataset, using our occlusion handling improves performance, increasing PTE6 from 93.02 to 93.21.

5) Finding the optimal number of Facial Regions

All standard landmark detectors, including LUVLi, output 68 landmarks across the face. Determining the number of regions to use, however, is a design choice that we explore. Using 48 regions creates multiple regions across the chin, right and left forehead, and right and left cheek as in Figure 2; to generate 96 regions, we split each of the 48 regions in half. We also

TABLE 9: Understanding effect of occlusion handling

Occlusion Handling?	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (%) ↑
✗	1.21±0.07	3.53±0.24	93.01±0.95
✓	1.17±0.11	3.46±0.21	93.21±1.14

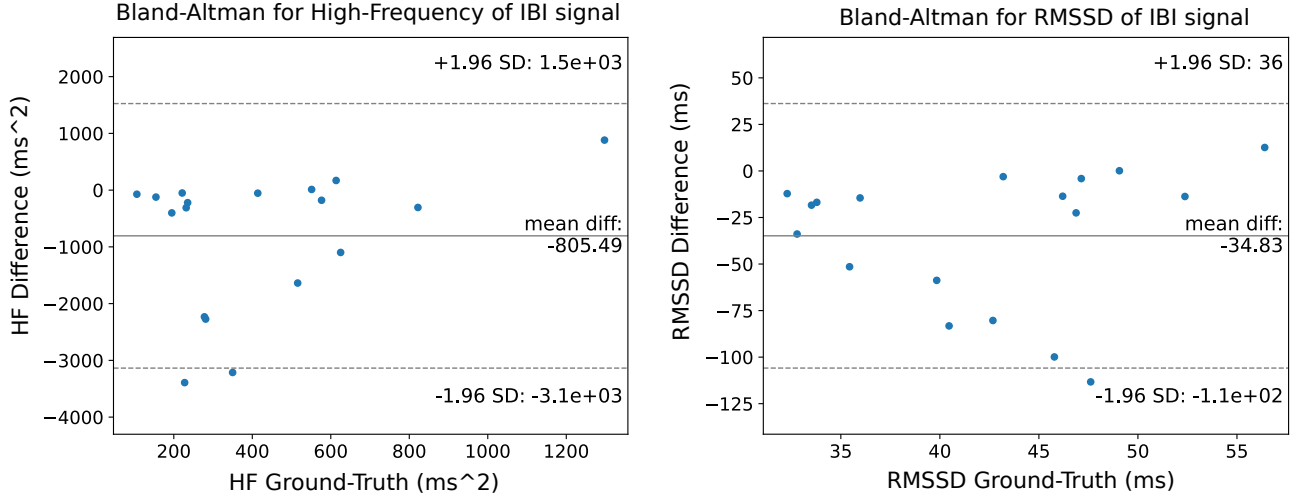


FIGURE 8: A Bland-Altman Analysis on the HF, and RMSSD metrics as described in Section V-A2 on all 1-min videos of the PURE test set.

TABLE 10: Comparing the 5-region TURNIP vs the 48-region TURNIP

Type	MAE (bpm) ↓	RMSE(bpm) ↓	PTE6 (%) ↑
TURNIP 5-region	1.35	3.53	92.24
TURNIP 48-region	1.04	2.84	93.02
TURNIP 96-region	1.23	3.33	92.79

TABLE 11: Effects of data augmentation and GRU on TURNIP performance using the MR-NIRP Car dataset (10-second windows) and MMSE-HR dataset (30-second windows).

Method		MR-NIRP Car				MMSE-HR	
Aug.	GRU	Driving		Garage		RMSE	PTE6
RMSE	PTE6	RMSE	PTE6	RMSE	PTE6	RMSE	PTE6
✗	✓	10.7	61.9	5.9	81.9	5.72	91.47
✓	✗	11.4	63.3	5.0	89.7	3.73	89.92
✓	✓	11.4	65.1	4.6	89.7	3.46	93.21

defined 5 regions which are super regions of the chin, right forehead, left forehead, right cheek, and left cheek.

TURNIP was re-trained in the 5-channel and 96-channel scenarios. We see optimal performance when using 48 regions in Table 10. We surmise that 5 regions are too few to capture robust signals when motion and illumination noise is high. Conversely, 96 regions are too small especially during the spatial averaging step in time-series extraction, resulting in pixel signals that are extremely sensitive to pixel outliers. Therefore, 48 regions are ideal.

6) Effects of Gated Recurrent Unit, Data Augmentation, and comparing against LSTM

Table 11 shows the effect of including or omitting the GRU component in our spatio-temporal denoising U-Net. We

clearly see that the GRU plays an important role in improving the performance. The table shows that our data augmentation generally improves results on the MR-NIRP Car dataset. We see that data augmentation improves the PTE6 on the MMSE-HR dataset and both subsets of the MR-NIRP Car dataset, and it improves RMSE on all but the the “Driving” subset of MR-NIRP Car. The data augmentation and GRU both help our method outperform the previous methods on the MR-NIRP Car dataset, as shown in Table 5.

We also compare our choice of the GRU, a simpler architecture, against another popular recurrent unit, the LSTM. We replace all GRU units in Figure 4 with LSTM units, and retrain TURNIP: our results are in Table 12. We achieve best performance when using TURNIP with the GRU. One potential reason this occurs is because of overfitting with LSTMs due to parameter counts. Consider an input size I and a hidden state size of H . In an LSTM there are **four** gates (input, forget, cell, and output), each of which has a weight matrix that transforms the input to the hidden size ($I \times H$, and the hidden state to the hidden size ($H \times H$). Then, there are biases for each gate (H). The number of parameters can be written as

$$\begin{aligned} \text{Parameters}_{\text{LSTM}} &= 4 \times ((I \times H) + (H \times H) + H) \\ &= 4 \times H \times (I + H + 1) \end{aligned} \quad (9)$$

The GRU only has **three** gates (update, reset, new hidden state). Considering an input size I and a hidden state size of H , for each gate we have a transformation from the input to the hidden size ($I \times H$, and the hidden state to the hidden size ($H \times H$). Then, there are biases for each gate (H). The number of parameters can be written as

$$\begin{aligned} \text{Parameters}_{\text{GRU}} &= 3 \times ((I \times H) + (H \times H) + H) \\ &= 3 \times H \times (I + H + 1) \end{aligned} \quad (10)$$

TABLE 12: Comparing performance on the MMSE-HR dataset when using GRUs and LSTMs

Type	MAE (bpm) ↓	RMSE(bpm) ↓	PTE6 (%) ↑
TURNIP w/LSTM	1.35	3.53	92.24
TURNIP w/GRU	1.04	2.84	93.02

TABLE 13: Evaluating ICA/CHROM/POS using our signal extraction pipeline on the MMSE-HR dataset

Method	MAE (bpm) ↓	RMSE (bpm) ↓	PTE6 (bpm) ↑
ICA [6]	5.44	12.00	-
ICA (our pipeline)	7.62	16.57	74.41
CHROM [17]	3.74	8.11	-
CHROM (our pipeline)	2.84	9.77	88.97
POS [18]	3.90	9.61	-
POS (our pipeline)	4.08	11.25	82.94
TURNIP	1.04	2.84	93.02

Therefore the LSTM has about 33% more parameters. We surmise that, while the performance with the LSTM is still very good, some of the parameters in the LSTM are overfitting the data, which results in worse performance. For this reason, we select the GRU.

7) Effects of our Signal Extraction on ICA/POS/CHROM

We evaluate the performance of ICA/POS/CHROM on our custom signal extraction pipeline (Face Detection + LUVLI Landmark Detection + Time-Series Extraction) and display the results in Table 13. We note a few differences in implementation: as compared to TURNIP which used 48 subregions (which were later collapsed into one through convolutional layers), the original ICA/CHROM/POS used the entire face as a single region for estimation. For our implementation of ICA/CHROM/POS, we extracted a 48 channel time series from the video; to capture the pulse rate, we converted each region into its power spectrum, summed each frequency bin across all 48 regions, and selected the bin with the highest power as our pulse rate estimate. We note that TURNIP has superior performance under the same signal extraction technique, showing the validity of our method.

8) Use of Correlation loss compared to traditional losses

For TURNIP, we were inspired by the work of [33] and chose to use the correlation loss to train our network. The Pearson correlation coefficient is suitable for aligning signals and their peaks in the time domain. This occurs because the correlation between two sinusoids is maximized when the signals are in-phase, which is the desired effect. However, various other losses have been used to train rPPG network. We compare the use of the correlation loss against the L_2 (MSE) loss and the L_1 loss in terms of heart rate estimation performance in Table 14. We see that all three methods generate similar performance, with the correlation loss providing the best performance in terms of MAE and RMSE. This indicates that

TABLE 14: Comparing TURNIP trained on three different losses

Type	MAE (bpm) ↓	RMSE(bpm) ↓	PTE6 (%) ↑
TURNIP w/ L_1 loss	1.12	3.21	94.57
TURNIP w/ L_2 loss	1.06	2.95	95.34
TURNIP w/Correlation loss	1.04	2.84	93.02

our model is not brittle to various algorithmic design choices and works well across the design space.

VI. CONCLUSION

In this paper, we build a modular pipeline for non-contact pulse rate estimation and pulse rate variability from facial videos that is composed of modules that perform face and landmark detection, time series extraction, and pulse signal estimation. Compared to previous end-to-end deep-network methods that map directly from the RGB frames to the final output, our modular algorithm significantly improves the estimation results. Our novel handling of self-occluded and out-of-frame landmarks in our time series extractor and TURNIP pulse signal estimator make our algorithm robust to varying levels of occlusion. Additionally, our signal model uses the ratio of pixel intensities of the red channel to the green channel for RGB videos, and leads to significant improvements across metrics. Our results demonstrate a new state-of-the-art in estimating pulse signals from facial videos, and our model achieves this using stages that are modular and interpretable. We have tested our algorithm in two different imaging domains (RGB and near-infrared), outperforming previous methods in both domains.

Future work should continue to address real-world remote pulse rate and pulse rate variability estimation with increasing variation in illumination and motion. While this work and many previous works address some of the fundamental issues associated with illumination variation and motion, further research could help make algorithms even more robust to these issues.

APPENDIX

In Table 7, we tested our algorithm on high-motion and low-motion splits of the PURE and MMSE-HR datasets. We enumerate the data splits below.

- **High-motion (PURE):** Videos labeled as 03-Slow Translation, 04-Fast Translation, and 06-Medium Rotation.
- **Low-Motion (PURE):** Videos labeled as 01-Steady, 02-Talking, and 05-Small Rotation
- **High-Motion (MMSE-HR):** F005-T10, F006-T11, F007-T11, F008-T10, F009-T11, F010-T11, F011-T11, F012-T11, F013-T8, F014-T8, F015-T8, F016-T8, F017-T8, F018-T1, F019-T11, F020-T10, F021-T11, F022-T10, F023-T10, F024-T10, F025-T10, F026-T10, F027-T10, M001-T11, M002-T11, M003-T10, M004-

T10, M005-T10, M006-T10, M007-T10, M008-T11, M009-T10, M010-T11, M011-T8, M012-T11, M013-T11, M014-T10, M015-T10, M016-T11, M017-T10

- **Low-motion (MMSE-HR):** F005-T11, F006-T10, F007-T10, F008-T11, F009-T10, F010-T10, F011-T10, F012-T10, F013-T10, F013-T11, F013-T1, F014-T10, F014-T11, F014-T1, F015-T10, F015-T11, F015-T1, F016-T10, F016-T11, F016-T1, F017-T10, F017-T11, F017-T1, F018-T10, F018-T11, F018-T8, F019-T10, F020-T11, F020-T1, F020-T8, F021-T10, F022-T11, F022-T1, F022-T8, F023-T11, F024-T111, F025-T11, F026-T11, F027-T11, M001-T10, M002-T10, M003-T11, M004-T11, M005-T11, M006-T10, M007-T11, M008-T10, M009-T11, M010-T10, M011-T11, M012-T10, M013-T10, M014-T11, M016-T11, M016-T10, M017-T11

REFERENCES

- [1] G. Nittari, D. Savva, D. Tomassoni, S. K. Tayebati, and F. Amenta, "Telemedicine in the covid-19 era: A narrative review based on current evidence," *International Journal of Environmental Research and Public Health*, vol. 19, no. 9, 2022. [Online]. Available: <https://www.mdpi.com/1660-4601/19/9/5101>
- [2] B. R. Bloem, E. R. Dorsey, and M. S. Okun, "The Coronavirus Disease 2019 Crisis as Catalyst for Telemedicine for Chronic Neurological Disorders," *JAMA Neurology*, vol. 77, no. 8, pp. 927–928, 08 2020. [Online]. Available: <https://doi.org/10.1001/jamaneurol.2020.1452>
- [3] W. Chen and D. McDuff, "Deepphys: Video-based physiological measurement using convolutional attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [4] E. M. Nowara, D. McDuff, and A. Veeraraghavan, "The benefit of distraction: Denoising camera-based physiological measurements using inverse attention," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 4955–4964.
- [5] M.-Z. Poh, D. J. McDuff, and R. W. Picard, "Advancements in noncontact, multiparameter physiological measurements using a webcam," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 7–11, 2011.
- [6] P. Ming-Zher, D. J. McDuff, and R. W. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," *Opt. Express*, vol. 18, no. 10, pp. 10762–10774, May 2010. [Online]. Available: <https://opg.optica.org/oe/abstract.cfm?URI=oe-18-10-10762>
- [7] X. Liu, Z. Jiang, J. Fromm, X. Xu, S. Patel, and D. McDuff, "Metaphys: Few-shot adaptation for non-contact physiological measurement," in *Proceedings of the Conference on Health, Inference, and Learning*, ser. CHIL '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 154–163. [Online]. Available: <https://doi.org/10.1145/3450439.3451870>
- [8] E. M. Nowara, T. K. Marks, H. Mansour, and A. Veeraraghavan, "Near-infrared imaging photoplethysmography during driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 4, pp. 3589–3600, 2022.
- [9] D. Shao, Y. Yang, C. Liu, F. Tsow, H. Yu, and N. Tao, "Noncontact monitoring breathing pattern, exhalation flow rate and pulse transit time," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 11, pp. 2760–2767, 2014.
- [10] P. H. Charlton, D. A. Birrenkott, T. Bonnici, M. A. F. Pimentel, A. E. W. Johnson, J. Alastruey, L. Tarassenko, P. J. Watkinson, R. Beale, and D. A. Clifton, "Breathing rate estimation from the electrocardiogram and photoplethysmogram: A review," *IEEE Reviews in Biomedical Engineering*, vol. 11, pp. 2–20, 2018.
- [11] Y. Cho, N. Bianchi-Berthouze, and S. J. Julier, "DeepBreath: Deep learning of breathing patterns for automatic stress recognition using low-cost thermal imaging in unconstrained settings," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, oct 2017. [Online]. Available: <https://doi.org/10.1109%2Facii.2017.8273639>
- [12] J. Fei and I. Pavlidis, "Thermistor at a distance: Unobtrusive measurement of breathing," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 4, pp. 988–998, 2010.
- [13] K. Iuchi, R. Miyazaki, G. C. Cardoso, K. Ogawa-Ochiai, and N. Tsumura, "Remote estimation of continuous blood pressure by a convolutional neural network trained on spatial patterns of facial pulse waves," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022, pp. 2138–2144.
- [14] E. M. Nowara, T. K. Marks, H. Mansour, and A. Veeraraghavan, "Sparseppg: Towards driver monitoring using camera-based vital signs estimation in near-infrared," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 1353–1359.
- [15] M. Lewandowska, J. Rumiński, T. Kociejko, and J. Nowak, "Measuring pulse rate with a webcam — a non-contact method for evaluating cardiac activity," in *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2011, pp. 405–410.
- [16] G. de Haan and A. van Leest, "Improved motion robustness of remote-ppg by using the blood volume pulse signature," *Physiological Measurement*, vol. 35, no. 9, p. 1913, aug 2014. [Online]. Available: <https://dx.doi.org/10.1088/0967-3334/35/9/1913>
- [17] G. de Haan and V. Jeanne, "Robust pulse rate from chrominance-based rppg," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2878–2886, 2013.
- [18] W. Wang, A. C. den Brinker, S. Stuijk, and G. de Haan, "Algorithmic principles of remote ppg," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 7, pp. 1479–1491, 2017.
- [19] X. Liu, J. Fromm, S. Patel, and D. McDuff, "Multi-task temporal shift attention networks for on-device contactless vitals measurement," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 19400–19411. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/e1228be46de6a0234ac22ded31417bc7-Paper.pdf
- [20] A. Comas, T. K. Marks, H. Mansour, S. Lohit, Y. Ma, and X. Liu, "TURNIP: Time-series U-net with Recurrence for NIR Imaging PPG," in *2021 IEEE International Conference on Image Processing (ICIP)*, 2021, pp. 309–313.
- [21] J. A. Crowe and D. Damianou, "The wavelength dependence of the photoplethysmogram and its implication to pulse oximetry," in *1992 14th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 6, 1992, pp. 2423–2424.
- [22] L. F. C. Martinez, G. Paez, and M. Strojnik, "Optimal wavelength selection for noncontact reflection photoplethysmography," in *22nd Congress of the International Commission for Optics: Light for the Development of the World*, vol. 8011. SPIE, 2011, pp. 2388–2394.
- [23] S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, and N. Sebe, "Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2396–2404.
- [24] Z. Sun and X. Li, "Contrast-phys: Unsupervised video-based remote physiological measurement via spatiotemporal contrast," in *European Conference on Computer Vision*, 2022.
- [25] Z. Yue, S. Ding, S. Yang, H. Yang, Z. Li, Y. Zhang, and Y. Li, "Deep super-resolution network for rppg information recovery and noncontact heart rate estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021.
- [26] S.-Q. Liu, X. Lan, and P. C. Yuen, "Remote photoplethysmography correspondence feature for 3d mask face presentation attack detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 558–573.
- [27] E. Lee, E. Chen, and C.-Y. Lee, "Meta-rppg: Remote heart rate estimation using a transductive meta-learner," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*. Springer, 2020, pp. 392–409.
- [28] M. Hu, F. Qian, D. Guo, X. Wang, L. He, and F. Ren, "Eta-rppgnet: Effective time-domain attention network for remote heart rate measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.
- [29] J. Du, S.-Q. Liu, B. Zhang, and P. C. Yuen, "Dual-bridging with adversarial noise generation for domain adaptive rppg estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 355–10 364.

- [30] X. Liu, Y. Zhang, Z. Yu, H. Lu, H. Yue, and J. Yang, "rppg-mae: Self-supervised pretraining with masked autoencoders for remote physiological measurements," *IEEE Transactions on Multimedia*, 2024.
- [31] R. Song, H. Chen, J. Cheng, C. Li, Y. Liu, and X. Chen, "PulseGAN: Learning to generate realistic pulse waveforms in remote photoplethysmography," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 5, pp. 1373–1384, 2021.
- [32] X. Niu, H. Han, S. Shan, and X. Chen, "Synrhythm: Learning a deep heart rate estimator from general to specific," in *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 3580–3585.
- [33] Z. Yu, X. Li, and G. Zhao, "Remote photoplethysmograph signal measurement from facial videos using spatio-temporal networks," 2019. [Online]. Available: <https://arxiv.org/abs/1905.02419>
- [34] J. Lin, C. Gan, and S. Han, "Tsm: Temporal shift module for efficient video understanding," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 7082–7092.
- [35] F. Shaffer and J. P. Ginsberg, "An overview of heart rate variability metrics and norms," *Frontiers in public health*, vol. 5, p. 290215, 2017.
- [36] L. Salahuddin, J. Cho, M. G. Jeong, and D. Kim, "Ultra short term analysis of heart rate variability for monitoring mental stress in mobile settings," in *2007 29th annual international conference of the IEEE engineering in medicine and biology society*. IEEE, 2007, pp. 4656–4659.
- [37] S. Aeschbacher, T. Schoen, L. Dörig, R. Kreuzmann, C. Neuhauser, A. Schmidt-Trucksäss, N. M. Probst-Hensch, M. Risch, L. Risch, and D. Conen, "Heart rate, heart rate variability and inflammatory biomarkers among young and healthy adults," *Annals of medicine*, vol. 49, no. 1, pp. 32–41, 2017.
- [38] H. J. Baek, C.-H. Cho, J. Cho, and J.-M. Woo, "Reliability of ultra-short-term analysis as a surrogate of standard 5-min analysis of heart rate variability," *Telemedicine and e-Health*, vol. 21, no. 5, pp. 404–414, 2015.
- [39] M. L. Munoz, A. Van Roon, H. Riese, C. Thio, E. Oostenbroek, I. Westrik, E. J. de Geus, R. Gansevoort, J. Lefrandt, I. M. Nolte et al., "Validity of (ultra-) short recordings for heart rate variability measurements," *PLoS one*, vol. 10, no. 9, p. e0138921, 2015.
- [40] F. Shaffer, S. Shearman, and Z. M. Meehan, "The promise of ultra-short-term (ust) heart rate variability measurements," *Biofeedback*, vol. 44, no. 4, pp. 229–233, 2016.
- [41] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "Faceboxes: A cpu real-time face detector with high accuracy," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*. IEEE, 2017, pp. 1–9.
- [42] A. Kumar, T. K. Marks, W. Mou, Y. Wang, M. Jones, A. Cherian, T. Koike-Akino, X. Liu, and C. Feng, "Luvli face alignment: Estimating landmarks' location, uncertainty, and visibility likelihood," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 8233–8243.
- [43] X. Niu, Z. Yu, H. Han, X. Li, S. Shan, and G. Zhao, "Video-based remote physiological measurement via cross-verified feature disentangling," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 295–310.
- [44] Z. Sun and X. Li, "Contrast-phys+: Unsupervised and weakly-supervised video-based remote physiological measurement via spatiotemporal contrast," 2024.
- [45] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.
- [46] Z. Zhang, J. M. Girard, Y. Wu, X. Zhang, P. Liu, U. Ciftci, S. Canavan, M. Reale, A. Horowitz, H. Yang, J. F. Cohn, Q. Ji, and L. Yin, "Multimodal spontaneous emotion corpus for human behavior analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [47] R. Stricker, S. Müller, and H.-M. Groß, "Non-contact video-based pulse rate measurement on a mobile service robot," *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pp. 1056–1062, 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:8529212>
- [48] R. Spetlik, V. Franc, J. Cech, and J. Matas, "Visual heart rate estimation with convolutional neural network," in *British Machine Vision Conference*, 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:52219725>
- [49] P. van Gent, H. Farah, N. Nes, and B. Arems, "Heart rate analysis for human factors: Development and validation of an open source toolkit for noisy naturalistic heart rate data," in *Proceedings of the 6th HUMANIST Conference*, 2018.
- [50] P. Van Gent, H. Farah, N. Van Nes, and B. Van Arems, "Analysing noisy driver physiology real-time using off-the-shelf sensors: Heart rate analysis software from the taking the fast lane project," *Journal of Open Research Software*, vol. 7, no. 1, pp. 1–9, 2019.
- [51] A. Zadeh, Y. Chong Lim, T. Baltrušaitis, and L.-P. Morency, "Convolutional experts constrained local model for 3d facial landmark detection," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 2519–2528.
- [52] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Openface: an open source facial behavior analysis toolkit," in *2016 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2016, pp. 1–10.
- [53] X. Liu, M. Zhang, Z. Jiang, S. Patel, and D. McDuff, "Federated remote physiological measurement with imperfect data," in *CVPRW 2022*, 2022.
- [54] Z. Yu, Y. Shen, J. Shi, H. Zhao, P. H. Torr, and G. Zhao, "Physformer: Facial video-based physiological measurement with temporal difference transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 4186–4196.
- [55] X. Liu, B. Hill, Z. Jiang, S. Patel, and D. McDuff, "Efficientphys: Enabling simple, fast and accurate camera-based cardiac measurement," in *2023 IEEE/CVF WACV 2023*, Jan 2023. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/WACV56688.2023.00498>
- [56] S.-Q. Liu and P. C. Yuen, "Robust remote photoplethysmography estimation with environmental noise disentanglement," *IEEE Transactions on Image Processing*, vol. 33, pp. 27–41, 2024.
- [57] H. Lu, H. Han, and S. K. Zhou, "Dual-gan: Joint bvp and noise modeling for remote physiological measurement," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 12 399–12 408.
- [58] J. Gideon and S. Stent, "The way to my heart is through contrastive learning: Remote photoplethysmography from unlabelled video," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 3995–4004.
- [59] Z. Yue, M. Shi, and S. Ding, "Facial video-based remote physiological measurement via self-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [60] M. Kumar, A. Veeraraghavan, and A. Sabharwal, "DistancePPG: Robust non-contact vital signs monitoring using a camera," *Biomedical optics express*, vol. 6 5, pp. 1565–88, 2015.



VINEET SHENOY is a PhD student in the Department of Electrical and Computer Engineering at Johns Hopkins University. He is broadly interested in Computer Vision and Machine Learning, with a current focus on object detection using domain adaptation and few-shot learning techniques. He has performed research in Spectrum Sharing at the National Institute of Standards and Technology (NIST), and Biometrics at the MITRE Corporation. Previously, he earned his B.S. in Electrical and Computer Engineering and Computer Science at Rutgers University New-Brunswick, where he received the James Leroy Potter Award for Original Investigation as well as the Chancellor's Leadership Award. He is a member of the Phi Beta Kappa and Tau Beta Pi honor societies.



SHAOJU WU received the B.E. degree in biomedical engineering from Shenzhen University, Shenzhen, China, in 2015, the M.S. degree in biomedical engineering from University of Florida, Gainesville, USA, in 2017, and the Ph.D. degree in biomedical engineering from Worcester Polytechnic Institute, Worcester, MA, USA, in 2022. He is currently a postdoctoral associate in the Department of Radiology at Boston Children's Hospital, Harvard Medical School, since September 2022.

His research interests lies in machine learning, computer vision, and medical image analysis. His current research is in the area of artificial intelligence (AI) for medical imaging. In this capacity, he collaborates with clinical radiologists and imaging scientists to solve clinically relevant and impactful imaging problems in the field of pediatric imaging.



TIM K. MARKS (Member, IEEE) received the AB degree in Physics from Harvard University in 1991, the MS degree in Cognitive Science from the University of California, San Diego in 2001, and the PhD degree in Cognitive Science from the University of California, San Diego (UCSD) in 2006. Dr. Marks is a Senior Principal Research Scientist and the Leader of the Computer Vision Team at the Mitsubishi Electric Research Laboratories (MERL), in Cambridge, MA, where he has worked

since 2008. From 2006–2008, he was a postdoctoral researcher in robotics at the UCSD Department of Computer Science, in collaboration with the computer vision group at the NASA/Caltech Jet Propulsion Laboratory. Before attending graduate school, he was a high school math and physics teacher and then a mathematics textbook editor for Houghton Mifflin. His current research interests span many topics in computer vision and machine learning. He has published over 60 papers in journals and conferences (including the IEEE Transactions on Pattern Analysis and Machine Intelligence, CVPR, ICCV, ECCV, IROS, and NeurIPS), as well as a college Number Theory textbook. He has been granted over 20 US patents.

...



ARMAND COMAS is a PhD student at Northeastern University, funded by the Dean's Fellowship. He received his B.Sc. (2017) and M.Sc. (2019) in Telecommunications Engineering from Universitat Politècnica de Catalunya (UPC - BarcelonaTech). His current research is on the topic of interpretability within Machine Learning, with applications in fields ranging from Computer Vision to Dynamic Systems. He is currently a research visitor at the Vision Lab at Johns Hopkins University,

on the topic of interpretability within generative models. In the past years, he has done research stays at MERL Computer Vision Group, Netflix, Adobe and Google.



SUHAS LOHIT (S'15, M'20) received the M.S degree in Computer Engineering in 2015 and Ph.D. degree in Electrical Engineering in 2019 from Arizona State University. He is currently a Principal Research Scientist at Mitsubishi Electric Research Laboratories in Cambridge, MA, USA, where he has worked since 2020. His research interests lie in machine learning, computer vision and computational imaging, and artificial intelligence.



HASSAN MANSOUR (S'99, M'09, SM'17) received the B.E. degree in computer and communications engineering from the American University of Beirut, Beirut, Lebanon, in 2003, and the M.A.Sc. degree in electrical and computer engineering and the Ph.D. degree in electrical and computer engineering from The University of British Columbia, Vancouver, BC, Canada, in 2005 and 2009, respectively. Between January 2010 and January 2013, he was a Postdoctoral Research Fellow

with the Department of Computer Science, the Mathematics Department, and the Department of Earth, Ocean, and Atmospheric Sciences, The University of British Columbia. He is currently a Senior Principal Research Scientist with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. His research interests are in inverse problems, compressed sensing, sparse signal reconstruction, image enhancement, and scalable video compression and transmission. His current research is focused on the design of efficient acquisition schemes and reconstruction algorithms for natural images, radar sensing, video analytics, and inverse scattering problems. Dr. Mansour is a member of the IEEE Computational Imaging Technical Committee and the IEEE Sensor Array and Multichannel Technical Committee. He is also an Associate Editor for the IEEE Transactions on Signal Processing.