The Way to my Heart is through Contrastive Learning: Remote Photoplethysmography from Unlabelled Video

John Gideon* Simon Stent*
Toyota Research Institute
Cambridge, MA, USA
{john.gideon, simon.stent}@tri.global

Abstract

The ability to reliably estimate physiological signals from video is a powerful tool in low-cost, pre-clinical health monitoring. In this work we propose a new approach to remote photoplethysmography (rPPG) – the measurement of blood volume changes from observations of a person's face or skin. Similar to current state-of-the-art methods for rPPG, we apply neural networks to learn deep representations with invariance to nuisance image variation. In contrast to such methods, we employ a fully self-supervised training approach, which has no reliance on expensive ground truth physiological training data. Our proposed method uses contrastive learning with a weak prior over the frequency and temporal smoothness of the target signal of interest. We evaluate our approach on four rPPG datasets, showing that comparable or better results can be achieved compared to recent supervised deep learning methods but without using any annotation. In addition, we incorporate a learned saliency resampling module into both our unsupervised approach and supervised baseline. We show that by allowing the model to learn where to sample the input image, we can reduce the need for hand-engineered features while providing some interpretability into the model's behavior and possible failure modes. We release code for our complete training and evaluation pipeline to encourage reproducible progress in this exciting new direction.[1]

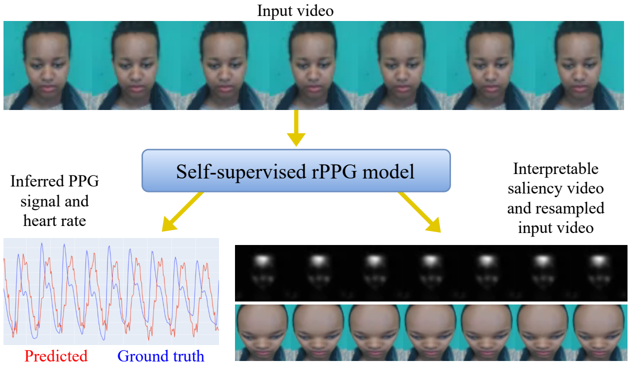
Input video
Self-supervised rPPG model
Inferred PPG signal and heart rate
Interpretable saliency video and resampled input video
Predicted
Ground truth

Figure 1. From a video of a person's face alone, our model learns to estimate the person's cardiac activity in the form of a photoplethysmographic (PPG) signal (left) observed through temporal patterns in the video, as well as a saliency signal (right) which shows where in the video the model's estimated activity is strongest (in this case, the center of the forehead). We show through extensive experiments on video datasets with physiological ground truth that our approach can match and sometimes even improve upon existing end-to-end supervised methods, while providing both *interpretability* into the model behavior and incurring *zero annotation cost* to train. The figure above shows a real output from our model trained on the UBFC [3] dataset. We note that the phase offset between predicted and ground truth PPG signals may be due to synchronization issues between the video and ground truth itself - a detail discussed further in Sec. 3.4.

1. Introduction

Understanding the physiological state of a person is important in many application areas, from health and fitness through to human resource management and human machine interaction. Conventional approaches to estimate such information, such as electrocardiograms (ECG) or photoplethysmograms (PPG), require interaction with the subject and are troublesome to setup, limiting their usefulness and scalability. In recent years, research utilizing advances from the field of computer vision and machine learning has explored and improved upon methods for passively monitoring physiological information from videos of subjects.

In this work we introduce a new method for remote photoplethysmography (rPPG), or imaging PPG, a technique in which changes in transmitted or reflected light from the body due to volumetric changes in blood flow are measured at a distance using a standard imaging device. This differs

*Equal contribution

[1] https://github.com/ToyotaResearchInstitute/RemotePPG

from the more intrusive contact PPG, in which the same signal is measured at peripheral body tissues such as the fingertips via a contact sensor which projects and measures reflected LED light. Compared to PPG, the signals for remote PPG are often too subtle for the human eye to perceive, but under certain illumination conditions, can be isolated and magnified in digital imagery if one knows where to look [34]. By finding these signals and using them to estimate underlying cardiac activity, particularly from webcam-quality video such as shown in the input video of Fig. 1, rPPG can therefore help to meet a need for low-cost, non-contact health monitoring.

In the field of computer vision, many researchers have tackled the problem of rPPG in the past, leaning on a wide variety of techniques from signal processing and machine learning (see e.g. [5, 17, 23, 33, 36]). Recent efforts have tended to favor deep learning, which is known for solving particular tasks well by discovering feature representations that are robust to many forms of nuisance variation. In rPPG, such variation takes the form of lighting changes, motion, and changes in facial appearance or gesture, all of which can easily obscure the underlying PPG signal. Supervised deep learning approaches to rPPG such as [5, 16, 23, 36] have shown that, with annotated data to train on, rPPG can be achieved with higher robustness to such variation. However, the cost of annotated data is not cheap, due to the need to equip subjects with contact PPG or ECG sensors while capturing data. It is therefore hard to scale the capture of such datasets, although, driven by data-hungry algorithms, there have been recent efforts in this direction [22, 23].

In this work we take a contrary approach to applying deep learning to rPPG. We view the problem through the lens of self-supervised learning, and in doing so bridge the data economy of older approaches with the robustness of learned representations. Our contrastive training approach is built around three assumptions about the underlying signal of interest:

A1 We assume the signal of interest lies within a certain range. We set this range for the rest of the paper at 40 to 250 beats per minute, which captures the vast majority of human heart rates [1].

A2 We assume the signal of interest typically does not vary rapidly over short time intervals: the heart rate of a person at time t is similar to their heart rate at $t + W$, where W is in the order of seconds.

A3 Finally, the signal has some visible manifestation (even if undetectable to the human eye) and is the dominant visual signal within the target frequency range.

Contributions. We show that by setting up a **contrastive learning** framework based on these assumptions, it is possible to train a deep neural network to estimate the PPG signal (and therefore track the heart rate) of a subject from video of their face, entirely without ground truth training data. We introduce novel loss functions for both supervised and contrastive training that are robust to desynchronization in the ground truth and take advantage of our above assumptions. Moreover, since the behavior of a deep neural network regressing PPG alone may be difficult to comprehend or have confidence in without access to annotated data, we propose a front-end **saliency-based sampling** module, inspired by [28], to accentuate the parts of the input data which are most relevant to a back-end PPG estimator. A by-product of this, as shown in Fig. 1, is that our model can also output interpretable saliency maps. These maps provides some transparency into the spatial location of the model's discovered signal of interest; in this case, the subject's forehead and parts of her nose and cheeks, which matches the conventional understanding of where the rPPG signal is strongest [14, 21]. We note that these contributions are independent but complementary. As shown in our experiments, the saliency sampler can be appended to both supervised and contrastive models with similar effect, and our contrastive model can learn to predict PPG without the saliency sampler. Finally, we unify existing freely available PPG video datasets and provide our complete training and evaluation pipeline to encourage further reproducible efforts in what we believe is an exciting direction of research in computer vision for human health monitoring.

2. Background and Related Work

Remote photoplethysmography. Approaches to rPPG, or heart rate (HR) estimation from video, can typically be broken down into three components: (i) a pre-processing stage, to minimize nuisance variation (for example through face detection and tracking) and discard irrelevant information in the input data; (ii) a PPG signal extraction stage; and (iii) a heart rate estimation stage from the estimated PPG signal. Early work in rPPG focused on finding signals within the image which were more easily accessible and perhaps more robust to nuisance variation, such as color over specific regions [15, 26, 27] or motion [2]. As dense facial tracking improved, the pre-processing part of the pipeline increased in complexity, incorporating techniques like landmark detection [17, 33], skin segmentation [3, 7] and carefully engineered ROI-based feature extraction [16, 23]. Of these, our work is most similar in spirit to [33], who describe an approach based on self-adaptive matrix completion to simultaneously estimate the heart rate signal and (learn to) select reliable face regions at each time. However, unlike their method, which requires keypoint tracking and careful image warping, ours is not necessarily face-specific and is arguably less aggressive in discarding potentially useful information: we do not reduce our feature space to chrominance features but instead pass the raw, warped image data to our PPG estimator.

In recent years, the reliance on relatively clean and stable input data has been slowly lifted as methods based on deep learning have increasingly proved themselves capable of learning more robustly through noise [5, 16, 22, 23, 35, 36, 30]. HR-CNN [30] uses a two-stage convolutional neural network (CNN) with a per-frame feature extractor and an estimator network. DeepPhys [5] uses a VGG-style CNN with separate predictions branches tailored towards motion and intensity. PhysNet [35] investigates both a 3DCNN based model and a model that combines 2DCNN and LSTM to learn spatio-temporal features. Yu *et al*. address the issue of rPPG detection in highly compressed facial videos by adding an additional autoencoder video enhancement stage to their model [36]. More recently, state of the art performance in rPPG has been achieved by a cross-verified feature disentangling strategy [23], which helps to isolate information which is most pertinent to physiological signal estimation. Their method involves computing hand-designed facial features called MSTmaps, which are average-pooled RGBYUV values across various combinations of regions of interest on the face. Their multi-branch output can be used to estimate both heart rate and PPG signal (as well as other possible signals such as respiratory frequency), with the joint loss from estimating both simultaneously returning further performance improvements. Finally, the RepNet model [6] demonstrated an effective way to estimate the period of repeated actions in video by computing the self-similarity of image representations. The authors show that, when applied to stable facial video preprocessed by [34], their model can recover human pulse.

As pointed out in the recent meta-RPPG work of Lee *et al*. [16], significant changes often occur in data distributions between model training and deployment, which can detrimentally affect the real-world performance of otherwise state-of-the-art end-to-end supervised learning approaches. To cope with such shifts, the authors propose a transductive meta-learner which can perform self-supervised weight adjustment from unlabelled samples. Our model is similar in intent, but rather than relying on modelling the domain shift, we allow for self-supervised training within entirely new domains from scratch. Crucially, since our approach still relies on an "end-to-end" trained deep neural network, it may be favorable compared to more traditional methods which require significant feature engineering or signal processing, since it is likely to improve with data.

Contrastive learning. To learn richer feature representations of data, contrastive learning [4, 9, 10, 11] proposes augmenting the data with different versions of itself during training, and contrasting a model's representations of the data in ways that encourage learning features with invariance or equivariance to particular augmentations. For example, in SimCLR [4], augmentations include spatial distortion (cropping, rotating, blurring) and chromatic distortion, and features can be learned which can help identify the semantic identity of objects in images with less sensitivity to such distortions. Unlike these techniques, in our case the signal of interest is stronger temporally than it is in an individual image, where it remains mostly imperceptible to the human eye. We therefore deliberately avoid image-domain augmentation, and focus instead on frequency augmentation. Specifically, by resampling video sequences at different rates and forcing the network to learn to detect whether two videos have similar or dissimilar underlying signals, we show that it is possible for the network to learn to filter particular signals of interest from the input data.

In the context of rPPG, one issue with using contrastive learning may be that, in the absence of annotated data to evaluate with, it is hard to understand model performance. To allow for some transparency into the model's inner workings, we equip it with a saliency sampling layer which shows which particular parts of the image were used by the system. Our approach builds on prior work [28], but applies the sampling layer to a self-supervised task and introduces additional priors to improve the behavior of the saliency map. While strictly not necessary for our approach to work, the sampler provides the system with an interpretable intermediate output, which can allow a practitioner to determine whether or not the network has converged to a sensible solution when training without ground truth data.

3. Method

An overview of our approach is shown in Fig. 2. We now describe each stage of the pipeline in sequence.

3.1. Preprocessing

For all datasets studied, we adopt a simple preprocessing procedure. We first estimate a bounding box around the face [37] and add an additional 50% scale buffer to the box before extracting a 192×128 frame. We only update the buffered bounding box location on subsequent frames if the new non-buffered bounding box is outside the larger one. This ensures relative stability of the video, while still allowing for occasional movement and re-alignment. Because the datasets used in this paper do not include full body or camera motion, re-alignment occurs infrequently. We finally crop and scale the videos to a 64×64 resolution for input to the model, which we found to return comparable performance to larger resolutions at lower computational cost.

3.2. Saliency Sampler

Input video sequences are passed to an optional saliency sampler module, building on work from [28]. In the context of our model, the module's purposes is two-fold: firstly, to

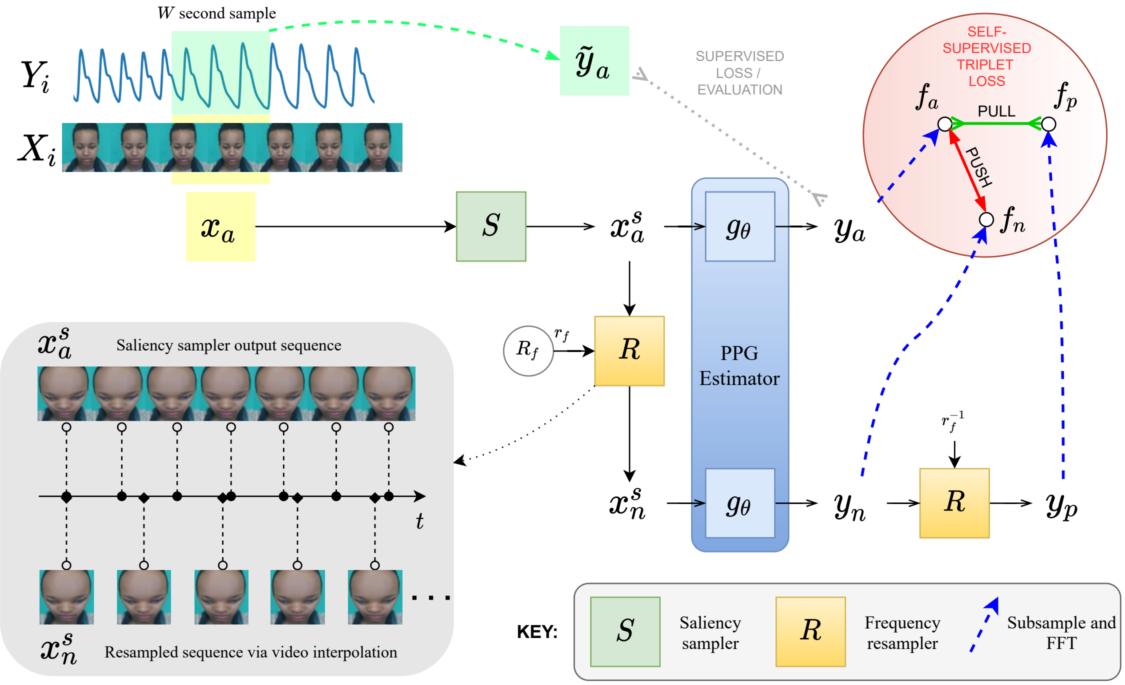

Figure 2. **Overview of our approach.** We first sample a video clip, x_a, of length W from the source video. This video is passed through the saliency sampler, S, to generate the warped anchor, x_a^s. The anchor is passed through a PPG Estimator g_θ to get y_a. If supervised training is employed, we employ a maximum cross-correlation (MCC) loss between the ground truth $\tilde{y}_a$ and y_a. If instead contrastive training is used, a random frequency ratio r_f is sampled from a prior distribution. The warped clip x_a^s is then passed through the frequency resampler R to produce the negative sample x_n^s, showing a subject with an artificially higher heart rate. This sample is passed through g_θ to produce the negative example PPG y_a. The negative sample is again resampled with the inverse of r_f to produce a positive example PPG y_p. Finally, the contrastive loss, MVTL, is applied to the PPG samples, using a PSE MSE distance metric. For further details about metrics and losses used, please see Sec. 3.4.

provide transparency as to what the PPG estimator is learning, which is particularly valuable if little annotated data exists to validate with; and secondly, to warp the input image to spatially emphasize task-salient regions, before passing it on to the task network (the PPG estimator) as described in [28]. In all experiments we use a pre-trained ResNet-18 [12], truncated after the conv2_x block, which we empirically find to perform well for the task without incurring significant computational overhead. Diverging from [28], we optionally impose two additional loss terms:

$$\mathcal{L}_{\text{sparsity}} = -\frac{1}{ND}\sum_{i}^{D}\sum_{j}^{N} s_i^j \log(s_i^j) \quad (1)$$

$$\mathcal{L}_{\text{temporal}} = \frac{1}{N(D-1)}\sum_{i}^{D-1}\sum_{j}^{N}(d_{i,i+1}^j)^2 \quad (2)$$

$$\mathcal{L}_{\text{saliency}} = w_s\mathcal{L}_{\text{sparsity}} + w_t\mathcal{L}_{\text{temporal}} \quad (3)$$

where s_i^j is the value of the per-frame softmax-normalized saliency map at frame i in the sequence and position j in the frame, $d_{i,i+1}^j$ is the difference between saliency pixel j from frame i to $i+1$, N is the number of pixels in a frame and D is the number of frames in the video sequence. The sparsity term favors solutions which have lower entropy (i.e. are spatially sparse, such as the forehead in Fig. 1), while the temporal consistency favors solutions that are smooth from frame to frame.

3.3. PPG Estimator

We use a modified version of the 3DCNN-based PhysNet architecture as our PPG estimator [35]. The core of PhysNet is a series of eight 3D convolutions with a kernel of (3, 3, 3), 64 channels, and ELU activation. This allows for the network to learn spatio-temporal features over the input video. Average pooling and batch normalization are also employed between layers. In the PhysNet paper, two transposed convolutions are used to return the encoded representation to the original length. However, we found that these introduced aliasing in the output PPG signal. We modify this part of the network to instead use upsampling interpolation (×4) and a 3D convolution with a (3, 1, 1) kernel. This upsampling step is repeated twice and removes the aliasing. Empirically, we found it to improve test RMSE by 0.2 bpm on average across datasets. Next, we perform adaptive aver-

Loss function	Assumptions	Used by
Negative max cross correlation (MCC)	HR is within a known frequency band (Assumption 2)	Supervised loss
Multi-view triplet loss (MVTL)	HR is stable within a certain window (Assumption 1)	Contrastive loss
Power spectral density mean squared error (PSD MSE)	HR is within a known frequency band (Assumption 2)	Distance metric
Irrelevant power ratio (IPR)	HR is within a known frequency band (Assumption 2)	Validation metric

Table 1. **Loss functions and metrics.** The losses used during training, the distance metric used for contrastive self-supervision, and the validation metric used during self-supervision. All supervised losses are also used as supervised validation metrics.

age pooling to collapse the spatial dimension and produce a 1D signal. A final 1D convolution is applied to convert the 64 channels to the output single channel PPG. Please see the supplementary material for full architectural details.

3.4. Loss Functions

We use a variety of loss functions and metrics during training, as summarized in Table 1.

We propose **maximum cross-correlation** (MCC) as a new loss function and metric for rPPG supervised training. While PC assumes PPG synchronization, MCC determines the correlation at the ideal offset between signals. This causes the loss to be more robust to random temporal offsets in the ground truth, assuming heart rate is relatively stable (Assumption 1). The authors of meta-RPPG [16] adopt a similar approach with their use of an ordinal loss. However, this requires the model to learn an ordinal regression instead of a raw PPG signal. MCC can be computed efficiently in the frequency domain, as follows:

$$\mathrm{MCC} = c_{pr} \times \mathrm{Max}\left(\frac{F^{-1}\{\mathrm{BPass}(F\{y\} \cdot \overline{F\{\hat{y}\}})\}}{\sigma_y \times \sigma_{\hat{y}}}\right) \quad (4)$$

We first subtract the means from each signal to simplify the calculation - resulting in y and $\hat{y}$. Cross-correlation is then calculated in the frequency domain by taking the fast Fourier transform (FFT) of the two signals and multiplying one with the conjugate of the other. To prevent circular correlation, we zero-pad the inputs to the FFT to twice their length. We apply a band bass filter by zeroing out all frequencies outside the range of expected heart rate (40 to 250 bpm), enforcing our Assumption 2. We then take the inverse FFT of the filtered spectrum and divide by the standard deviation of the original signals, σ_y and $\sigma_{\hat{y}}$, to get the cross-correlation. The maximum of this output is the correlation at the ideal offset. We scale the MCC by a constant, c_{pr}, which is the ratio of the power inside the heart rate frequencies. This ensures the MCC is unaffected by the frequencies outside the relevant range. We use MCC as our loss function for supervised training. In the supplementary material we include an analysis of the robustness of MCC to randomly injected ground truth synchronization error, showing its more stable performance compared to more standard losses such as Pearson's correlation and the signal-to-noise ratio.

We also introduce **multi-view triplet loss** (MVTL) as our loss function for contrastive training. As shown in Fig. 2, our self-supervised pipeline has three output branches - anchor (y_a), positive (y_p), and negative (y_n). From these three branches, we take V_N subset views of length V_L. This enforces Assumption 1 - that heart rate is relatively stable within a certain window. Because of this, the signal within each view should appear similar. We then calculate the distance between all combinations of anchor and positive views (P_{tot}) and all combination of anchor and negative views (N_{tot}). We calculate $P_{tot} - N_{tot}$ and scale by the total number of views, V_N^2, to get the final loss.

We use the **power spectral density mean squared error** (PSD MSE) as the distance metric between two PPG signals when performing contrastive training with MVTL. We first calculate the PSD for each signal and zero out all frequencies outside the relevant heart rate range from 40 to 250 bpm (Assumption 1). We then normalize each to have a sum of one and compute the MSE between them.

Finally, we use the **irrelevant power ratio** (IPR) as a validation metric during contrastive training. We first calculate the PSD and split it into the relevant (40 to 250 bpm) and irrelevant frequencies (Assumption 2). We then divide the power in the irrelevant range with the total power. IPR can be used as an unsupervised measure of signal quality.

3.5. Training

Sampling. When training, we randomly sample W seconds from a video X_i and its associated physiological ground truth Y_i. For our experiments, we set W to ten seconds. We denote these subset clips as x_a and $\tilde{y}_a$, respectively. We randomly augment our training sets by artificially stretching shorter video clips to W seconds using trilinear interpolation. We use linear interpolation to mimic this stretching in the ground truth PPG and scale ground truth HR appropriately. At most, this effectively decreases the HR by 33%. If the calculated IPR for a given sample PPG exceeds 60%, we redraw a new W second subset.

Heart Rate Calculation. Given a PPG, we calculate heart rate by (1) zero-padding the PPG signal for higher frequency precision, (2) calculating the PSD, and (3) locating the frequency with the maximum magnitude within relevant heart rate range. We use this method to both calculate missing HR ground truth and HR from predicted PPG. We chose a simple PSD-based method instead of a learned one to maintain determinism.

Saliency Sampler. If enabled, the input video x_a is first passed through the saliency sampler (S). The resulting spatially warped video is denoted as x_a^s. The output of the saliency sampler can be used to verify the performance of the network, as explored in Sec. 5.2.

Supervised Training. When performing supervised training, only the top portion of Fig. 2 is used. The input video clip x_a^s is passed through the PPG Estimator g_θ, producing the PPG estimate y_a. We then apply the selected supervised loss function between $\tilde{y}_a$ and y_a.

Contrastive Training. When performing contrastive training, we randomly choose a resampling factor R_f between 66% and 80%. We then pass the anchor video clip x_a^s through the trilinear resampler R to produce the negative sample x_n^s. This effectively increases the frequency of the heart rate by a factor of 1.25 to 1.5. Both x_a^s and x_n^s are passed through the PPG Estimator g_θ, producing y_a and y_n, respectively. We then resample y_n using the inverse of R_f to output the positive signal y_p, whose frequency should match y_a. Finally, we apply the contrastive loss function, MVTL, using the PSD MSE distance. For our experiments, we set the number of views (V_N) to four and the length (V_L) to five seconds. As this method is unsupervised, we also use the validation set for training.

Further Training Details. We implemented our models using PyTorch 1.7.0 [25] and trained each model on a single NVIDIA Tesla V100 GPU. We used a batch size of 4 for all experiments and set w_s and w_t to 1, unless otherwise stated. In all experiments we use the AdamW optimizer with a learning rate of 10^{-5} and train for a total of 100 epochs. After supervised training, we select the model from the epoch with the lowest validation loss. Because the contrastive training does not use labels, we instead choose the model with the lowest IPR on the training set.

4. Datasets

To test our model as fairly as possible, we evaluated it on four publicly available rPPG datasets from recent literature. Table 2 shows the datasets which we considered for evaluation. During the construction of this table, it became clear that much prior work had evaluated their methods using combinations of proprietary datasets and/or datasets which were not freely available to industrial researchers. This made replication of results expensive or impossible. To avoid this, we opted to use freely available data for both training and evaluation. We ingested the following four datasets into a common data format, which we make available for other researchers to utilize.

Dataset	PPG	Subj.	Dur. (hrs)	Freely avail.
COHFACE [13]	✓	40	0.7	✓
ECG-Fitness [30]		17	1.7	✓
MAHNOB [29]	✓	27	9.0	
MMSE-HR [33]		40	0.8	
MR-NIRP-Car [24]	✓	19	3.3	✓
MR-NIRP-Indoor [19]	✓	12	0.6	✓
OBF [36]	✓	106	177.0	
PURE [31]	✓	10	1.0	✓
UBFC-rPPG [3]	✓	42	0.8	✓
VIPL-HR [22]	✓	107	20.0	
VIPL-HR-V2 [18]	✓	500	21.0	

Table 2. **Survey of published datasets for rPPG analysis.** Not all datasets are available for open research. For our experiments, we selected the highlighted subset of RGB datasets containing a ground truth PPG signal. We did not use ECG-Fitness because it has extreme motion and utilizes ECG instead of PPG. We used MR-NIRP-Car in place of MR-NIRP-Indoor, as it was the more challenging set, containing body motion and lighting changes.

PURE [31] consists of 10 subjects (8 male, 2 female) performing different, controlled head motions in front of a camera (steady, talking, slow translation, fast translation, small rotation, medium rotation) for one minute per sequence, under natural lighting. During these sequences, the uncompressed images of the head, as well as reference PPG and heart rate from a finger clip pulse oximeter were recorded. The first two samples of the PPG were corrupted and were discarded during analysis. PURE contains predefined folds for training, validation, and test, and we use these to be comparable to related work. We run each experiment 25 times with different random seeds and average our performance.

COHFACE [13] consists of 160 one minute videos from 40 subjects, captured under studio and natural light and recorded with a Logitech HD Webcam C525 and contact rPPG sensor. The videos are heavily compressed using MPEG-4 Visual, which was noted by [20] to potentially cause corruption of the rPPG signal. Similarly to PURE, the dataset comes with preassigned folds and we run each experiment 25 times for stability.

MR-NIRP-Car [24] is the first publicly available video dataset with ground truth pulse signals captured during driving. Data was captured simultaneously in RGB and near-infrared (NIR), with associated pulse oximeter recordings. The dataset contains 190 videos of 19 subjects captured during driving as well as inside a parked car in a garage. Each subject performed different motion tasks (looking around the car, looking at mirrors, talking, laughing, sitting still). To be consistent with [24], we only consider the subset of “RGB garage recordings”, which have minimal head motion and consistent lighting. One sample had to be discarded due to a corrupted compressed file. Furthermore, we noticed that there were many stretches of zero values in the

	PURE			COHFACE			MR-NIRP-Car			UBFC		
Method	RMSE	MAE	PC	RMSE	MAE	PC	RMSE	MAE	PC	RMSE	MAE	PC
Mean	16.0	13.0	-0.27	15.4	12.0	-0.19	15.1	12.7	-	21.8	19.1	0.00
Median	22.0	19.2	-0.05	17.8	14.6	-0.09	16.9	14.7	-	21.6	18.8	-0.11
HR-CNN [30]	**2.4**	**1.8**	0.98	10.8	8.1	0.29	-	-	-	-	-	-
Nowara et al. [24]	-	-	-	-	-	-	*2.9	-	-	-	-	-
Meta-rppg [16]	-	-	-	-	-	-	-	-	-	*7.4	*6.0	*0.53
Our Supervised	2.6	2.1	**0.99**	7.8	2.5	0.75	**1.6**	**0.7**	0.96	4.9	3.7	**0.95**
With Saliency	2.6	2.1	**0.99**	7.6	2.3	0.76	1.8	0.8	**0.97**	5.0	3.8	**0.95**
Our Contrastive	2.9	2.3	**0.99**	**4.6**	**1.5**	**0.90**	4.1	1.7	0.91	**4.6**	**3.6**	**0.95**
With Saliency	3.0	2.3	**0.99**	5.5	1.8	0.84	5.2	2.4	0.87	6.1	5.0	0.91

Table 3. **Experiment Results.** Results on all datasets using our supervised and contrastive systems, with and without saliency, averaged over 25 independent training runs. We compare with mean and median baselines, as well as the strongest comparable baseline we could find for each dataset. Overall, our contrastive approach performs on par or better than existing methods for most reported results, and in some cases better than our own supervised training. *Note that, for reasons described in Sec. 4 and Sec. 5.1, baselines for MR-NIRP-Car and UBFC are not directly comparable to our results. We make our full data pre-processing and evaluation pipeline available to support fair comparisons in the future.

PPG signal that we had to resample around during training. Due to the lack of a set of folds, we split the dataset into five folds by subject id. We then conducted five training runs using a different held-out test set each time. We repeat this process five times, for a total of 25 runs per model, and then averaged the results across all runs.

UBFC-rPPG [3] contains uncompressed videos from 42 subjects, with ground truth PPG and heart rate data from a pulse oximeter. Heart rate variation was induced in participants by engaging them in a time-sensitive mathematical puzzle. To match the results of [16], for the purposes of testing we discard subjects 11, 18, 20, and 24, who were observed to have erroneous (< 5 bpm) heart rate data. However, since only the PPG data and not the videos are corrupted, we include those videos for self-supervised training. Because no fold splits are included in the UBFC dataset, we use the same test strategy as with MR-NIRP-Car.

Commonalities. In all datasets, videos are captured at 30Hz. Where necessary, we interpolated the physiological data to synchronize it to get one sample for each video frame. Because we make the key assumption that heart rate does not vary over short time intervals (Assumption 2), we analyzed each of the datasets for the amount of variation in heart rate. In general, we found that heart rate did not vary by more than 2.5 bpm in the majority of data over a $10s$ window. Please see the supplementary material for further analysis.

5. Experiments

5.1. Dataset Performance

We first compare our method on a set of four recent PPG datasets in Table 3. As described in the previous section, while several larger datasets have recently been released [22, 23], we were unable to access them for benchmarking due to usage restrictions. We show the results of both our supervised and contrastive systems, with and without the use of a saliency sampler. We compare against two baselines which predict the mean and median of the test data. We calculate the root mean squared error (RMSE), mean absolute error (MAE), and Pearson's correlation (PC) of the predicted versus ground truth heart rate. We present further performance statistics over the PPG signals in the supplementary material.

PURE. The results across all systems, including the HR-CNN baseline [30], are similar for PURE - between 2.4 and 3.0 RMSE. This close-to-ideal performance is likely due to the high quality video, constrained environment, and minimal movement. Notably, the contrastive method is able to achieve similar results without the use of ground truth.

COHFACE. We find that our contrastive system performs best on COHFACE, with an RMSE of 4.6, despite not using labels during training. This provides evidence that our system is more robust to video compression versus other methods. While our supervised model performs worse (7.8 RMSE), it still outperforms the strongest comparable baseline (10.8 RMSE) [30].

MR-NIRP-Car. To be comparable to [24], we report the RGB garage minimal motion subset. Although the baseline is already low at 2.9 RMSE, our supervised system improves this to 1.6 RMSE. Note that the baseline was knowledge-based and did not rely on training data, making the results not entirely comparable. Our contrastive system performs slightly worse (4.1 RMSE), but does so without ground truth labels.

UBFC. Lastly, we consider the UBFC dataset, compar-

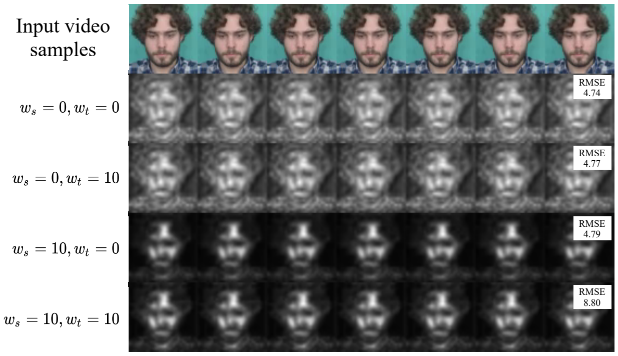

Figure 3. **Effect of regularization on the computed saliency estimates of a contrastively trained network.** The effect of the sparsity regularizer is especially clear. We note that the rPPG model performance was observed to be stable under various regularization settings, although training became empirically less stable at higher levels of regularization (as in the bottom row, caused by one of five training runs failing to converge).

ing against the 7.4 RMSE baseline from [16]. Note they instead use the first two seconds of all samples for adaptation, so the results are not perfectly comparable. Our supervised (4.9 RMSE) and contrastive (4.6 RMSE) methods both achieve a similar, improved performance.

5.2. Saliency Sampler

We evaluate our saliency sampler both quantitatively (whether or not it changes the PPG estimator's performance on the primary task of rPPG), and qualitatively (whether or not it aids in interpretation of the model's behavior). Results from adding in the saliency sampler to the model when training for both supervised and contrastive models are shown in Table 3. In PURE, the sampler had no significant effect on performance, while for COHFACE, MR-NIRP-Car, and UBFC it alters performance by at most one point. In Fig. 3 we show the qualitative effect of varying the sparsity and temporal regularization parameters during training. We include an experiment looking at the sensitivity to these parameters in the supplementary material.

We conjecture that while the temporal regularizer may particularly help performance in videos with little motion – by allowing motion cues to pass through to the PPG estimator – in videos with large motion it may hinder performance. On the other hand, the sparsity regularizer reliably helps to achieve intepretable saliency maps without harming performance significantly.

To illustrate the qualitative value of the sampler, we run a toy experiment injecting a periodic nuisance signal, as shown in Fig. 4. Under contrastive training, the sampler can be used to determine that the PPG estimator has found a spurious signal, and is not working as expected. Under supervised training, the sampler learns to remove the periodic signal from the input data to the PPG estimator altogether.

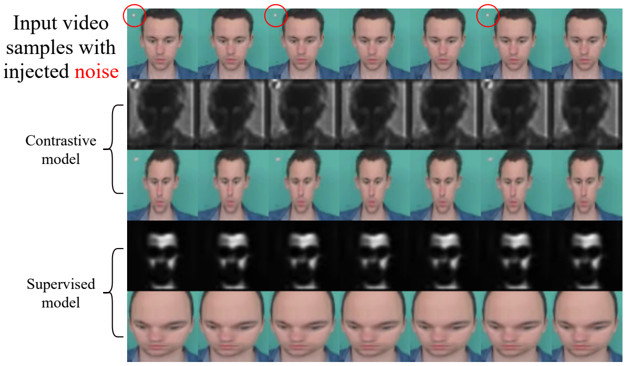

Figure 4. **An example of interpretable model behavior.** We added a random flashing pixel block (highlighted in the top row) at 60-180 bpm to the UBFC dataset and trained both our contrastive model and our supervised model. In the contrastive case, without the need for ground truth validation data, the saliency map reveals that the model has learned to use the noise signal rather than learn the signal of interest on the subject. In contrast, in the supervised setting, the saliency sampler learns to emphasize the skin of the subject, and completely discards the injected noise signal after resampling, since it has no relevance to the prediction of PPG.

6. Discussion

In this paper we presented a contrastive approach to estimating cardiac activity in a person from video of their face. We believe this is the first time that deep neural networks have been applied to the problem of remote photoplethysmography in a fully self-supervised manner, at zero annotation cost. This would allow heart rate detection to be adapted to a specific domain without first acquiring labelled data in the domain. We demonstrate the value of this approach through an accompanying workshop challenge submission [8], in which self-supervised training on domain-shifted test data was shown to improve system performance.

In addition, we introduced a novel loss for supervised training that is more robust to ground truth synchronization error and yields improved performance. We also proposed the use of a saliency sampler to provide interpretable output to confirm whether the system is behaving as expected.

Our work opens the door to training on significantly larger, unlabelled datasets, such as sourced from the internet. This may help to improve the generalizability of heart rate estimation to more challenging domains and conditions, such as datasets with more severe lighting changes and head motion. We also aim to further explore the use of learned saliency or attention-like mechanisms to more efficiently direct the efforts of a downstream PPG estimator, in a way that better conserves the original raw image pixels.

Acknowledgments. We wish to thank Luke Fletcher for quickening our pulses at the right time, and the team at the Shapiro Cardiovascular Center at BWH from the bottom of (one of) our hearts.

ENCODER	in	out	kernel	stride	pad
Conv3d + BN3d + ELU	C	32	(1,5,5)	1	(0,2,2)
AvgPool3d			(1,2,2)	(1,2,2)	0
Conv3d + BN3d + ELU	32	64	(3,3,3)	1	(1,1,1)
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
AvgPool3d			(2,2,2)	(2,2,2)	0
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
AvgPool3d			(2,2,2)	(2,2,2)	0
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
AvgPool3d			(1,2,2)	(1,2,2)	0
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
Conv3d + BN3d + ELU	64	64	(3,3,3)	1	(1,1,1)
DECODER					
Interpolate			(2,1,1)		
Conv3d + BN3d + ELU	64	64	(3,1,1)	1	(1,0,0)
Interpolate			(2,1,1)		
Conv3d + BN3d + ELU	64	64	(3,1,1)	1	(1,0,0)
AdaptiveAvgPool3d			(-,1,1)		
Conv3d	64	1	(1,1,1)	1	(0,0,0)

Table 4. **Modified PhysNet-3DCNN architecture**. The architecture follows an encoder-decoder structure with 3D convolutions to represent patterns through time; "s" corresponds to stride, "p" to padding, "C" to the number of input channels.

SALIENCY NET	in	out	kernel	stride	pad
Conv2d + BN2d + ReLU	C	64	(1,7,7)	2	3
MaxPool			(1,3,3)	2	1
BasicBlock	64	64		1	
BasicBlock	64	64		1	
BasicBlock	64	64		1	

Table 5. **Saliency Network**. The architecture follows a Resnet-18 structure, truncated after `layer1`, with pre-trained ImageNet weights. Each BasicBlock consists of a 3 × 3 convolution, 2D batch normalization, ReLU, 3 × 3 convolution, 2D batch normalization, addition with the BasicBlock input (the residual) and a final ReLU. For further details see [12].

A. Appendix

A1. Model Architecture

For the PPG estimator we use a modified 3D-CNN version of PhysNet [35] as described in Table 4. Our modification is to use interpolation and convolution in the decoder instead of transposed convolution, which we found to reduce the aliasing artifacts that were present in the original model. For the saliency sampler, we use the architecture described in [28], swapping out the saliency network for the shallower model shown in Table 5 which was found to be sufficient for detecting facial parts (by the nature of the saliency maps learned).

A2. Other Loss Functions/Metrics

Pearson's correlation (PC) is commonly used as a loss and metric in other rPPG works (*e.g.* [35]). While it is scale invariant, it assumes that there is perfect temporal synchronization between the ground truth and observed data. Otherwise the network must be capable of learning a temporal offset, assuming the offset is constant.

Signal-to-noise ratio (SNR) is another baseline used in prior rPPG works (*e.g.* [30]) which train to match a ground truth heart rate instead of the full PPG signal. It relaxes the alignment assumption by expressing the loss in the frequency domain using the power spectral density (PSD). It calculates the amount of power in the target heart rate frequency bin of the PSD and compares it to the total other power in the PPG signal. Because of this, it assumes that all other frequencies should be zeroed out, which may remove meaningful harmonics.

A3. Dataset PPG Performance

To supplement Table 3 from the main paper, we present further results on the four PPG datasets in Table 6, using metrics which capture statistics of predicted vs. ground truth PPG signals (as opposed to the final predicted heart rate). We again show the results of both our supervised and contrastive systems, with and without the use of a saliency sampler. However, as these PPG statistics are not given for the other cited baseline systems, we instead provide only the mean and median methods as baselines.

We calculate the Pearson's correlation (PC), Max cross-correlation (MCC), and Signal-to-noise ratio (SNR) of the predicted versus ground truth PPG signals. While we present PC for comparison, we expect MCC and SNR to be better measures of system performance, as they are calculated in the frequency domain. This makes them more robust to desynchronization between predictions and ground truth. Supervised training uses MCC as a loss function and validation metric, while contrastive training works without guiding ground truth. Because of this, it is possible for either system to learn a random phase offset, as long as the overall frequency information is predictive of heart rate.

PC. Across all dataset results, we note that supervised training attains the highest PC. Even though desynchronization is not penalized during training, the model likely learns the easiest mapping between video and ground truth - one without an additional offset. This could indicate that the datasets only have minimal offset between observation and ground truth.

MCC. We also note that supervised training tends to result in higher MCC, which is likely due to the guiding ground truth. Without ground truth, the contrastive method is only able to learn the periodic signal visible in the input video. The supervised method would be able to learn to

	PURE			COHFACE			MR-NIRP-Car			UBFC		
Method	PC	MCC	SNR	PC	MCC	SNR	PC	MCC	SNR	PC	MCC	SNR
Mean	-0.01	0.12	-9.6	-0.01	0.14	-9.2	-0.02	0.36	5.6	0.00	0.10	-13.0
Median	0.00	0.08	-10.9	0.00	0.14	-9.2	0.00	0.30	1.4	0.01	0.10	-13.5
Our Supervised	0.54	**0.90**	18.1	0.23	0.57	15.4	**0.52**	**0.79**	17.8	**0.17**	**0.64**	13.3
With Saliency	**0.58**	**0.90**	18.1	**0.30**	0.57	15.5	0.42	0.78	17.9	0.15	**0.64**	13.3
Our Contrastive	0.02	0.79	19.4	-0.19	**0.65**	17.9	0.18	0.74	**19.3**	0.03	0.63	**14.3**
With Saliency	0.00	0.80	**19.5**	-0.04	**0.65**	**17.9**	0.27	0.74	18.8	0.01	0.61	13.3

Table 6. **Experiment PPG Statistics.** PPG statistics on all datasets using our supervised and contrastive systems, with and without saliency. We compare with both a mean and median baseline. The top performing system varies greatly depending on the dataset and statistic. However, the contrastive systems often perform comparable or better than the supervised ones when considering sync-robust metrics (without the need for ground truth).

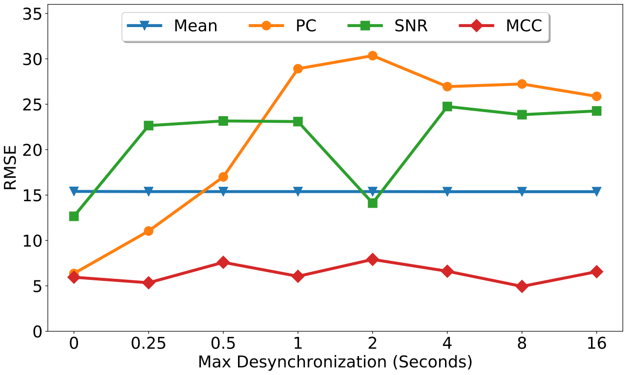

Figure 5. **Loss Function Robustness to Desynchronized Ground Truth.** The performance of supervised training on COHFACE with varying amounts of random desynchronization applied between the video and the ground truth PPG signal. We show the performance of 3 loss functions (PC, SNR, MCC) against a Mean model baseline (which estimates the mean heart rate in the test set). Unlike other loss functions, MCC is shown to be robust to ground truth desynchronization.

replicate any repeating artifacts of the biometric PPG sensor, producing a stronger MCC. However, the overall performance on the contrastive COHFACE model is substantially better than that of the supervised one, as seen in the heart rate results. This likely lessens the relative impact of PPG artifacts, when compared with other datasets with closer performance.

SNR. Unlike MCC, SNR penalizes the learning of all other frequencies besides heart rate. So perfect PPG prediction can result in lower SNR performance, if the PPG includes other frequencies. Because the supervised training uses MCC as a loss function and the ground truth isn't a perfect sine wave, this encourages sub-optimal SNR. We see this reflected in the results, with contrastive learning having a higher SNR across all datasets.

A4. Loss Function Robustness

In Section A3, we examined how the metrics PC, MCC, and SNR can be used to gauge the performance of PPG prediction versus ground truth. In this section, we examine how each metric performs as a loss function during supervised training, as well as the impact of desynchronization between the observed video and the ground truth PPG.

We use the COHFACE dataset and compare versus a baseline that always predicts the mean heart rate in the test set. We select the maximum amount of injected synchronization error ($O_{\max}$) to range between 0 and 16 seconds. Each time a new clip is drawn during training, a random offset is chosen between $-O_{\max}$ and $O_{\max}$ using uniform sampling. We then shift the ground truth PPG by the selected offset using neighboring data. We train a model using the supervised pipeline and the selected loss function. We calculate the selected loss on held-out validation data each epoch and use the model with the lowest loss at test time. In this experiment, we do not use the saliency sampler since the purpose is to explore the robustness of supervised loss functions.

Figure 5 shows the RMSE performance of our supervised system for different $O_{\max}$ and loss functions. Without injected desynchronization, we find that PC and MCC perform similarly. This likely indicates that ground truth in COHFACE is consistently aligned with the video, either with a minimal or constant (learnable) offset. Because MCC is the offset-adjusted version of PC, we note that they have similar performance without the presence of offsets. However, when increasing amounts of synchronization error are applied, the performance of the system trained with PC quickly degrades, while the one trained with MCC remains relatively stable. We also note that SNR has consistently poor performance at all offsets, indicating that it is a weaker supervisory signal for rPPG compared to correlation-based measures. Based on these results, we favored MCC as a loss function for supervised training,

	RMSE change (↓)				MAE change (↓)				PC change (↑)			
$w_s \setminus w_t$	0	0.1	1	10	0	0.1	1	10	0	0.1	1	10
0	-	1.4	1.4	-0.9	-	0.7	0.6	-0.2	-	-0.06	-0.06	0.02
0.1	3.1	0.8	-0.8	0.1	1.3	0.4	-0.2	0.0	-0.18	-0.04	0.02	0.00
1	-0.8	0.0	-0.7	-0.4	-0.2	0.0	-0.2	-0.1	0.02	0.00	0.02	0.01
10	0.2	0.6	0.4	-0.9	0.0	0.2	0.2	-0.2	0.00	-0.02	-0.01	0.03

Table 7. **Sensitivity of saliency sampler regularization on rPPG performance for a contrastively trained model on UBFC.** As with Table 3 from the main paper, we average the results of five runs on different folds with held-out subjects, and report the performance differences relative to the zero regularization case, (w_s, w_t) = (0,0). rPPG performance tends to improve when the regularization parameters are set in the range [1,10], although the best parameters depend on variables such as dataset and resolution.

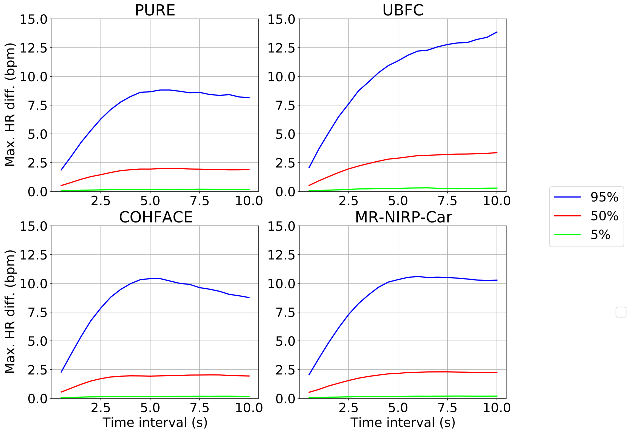

Figure 6. **Distribution of heart rate differences at different time intervals apart**. The 5%, 50% and 95% quantile lines are shown.

particularly if synchronization issues were suspected to be present in the training data.

A5. Dataset Statistics

Our approach relies on the assumption (Assumption 2 in the main paper) that "the signal of interest typically does not vary rapidly over short time intervals: the heart rate of a person at time t is similar to their heart rate at $t + W$, where W is in the order of seconds." In Fig. 6, we show the distribution of heart rate variation at different time intervals within each of the four datasets. With approximately 80%-100% certainty, depending on the dataset, one can assume that the heart rate at $t + 10$s is within 10bpm of the heart rate at time t. All datasets consistently have a median variation of about 2.5bpm over a 10s period. UBFC was found to contain the most heart rate variability of the datasets examined.

A6. Unsupervised Learning Protocol

In our contrastive experiments, since our method does not utilize ground truth labels, we fold validation data into the training set. In other words, in Table 3 of the main paper, the contrastive models "see" a little more training data than the supervised models. When validation data is excluded from contrastive training, we found test RMSE to worsen by 0.7 bpm on average across datasets. This shows the value of larger training sets when training with contrastive loss. Exploring the trade off between training data size and model performance is a topic for future work.

A7. Additional Baseline

While we selected the strongest comparable baseline we could find for each dataset, one reviewer requested the addition of the Siamese CNN proposed by [32]. However, we note that this work is not directly comparable for several reasons, in particular: (i) for UBFC evaluation, we do not pre-train our model on COHFACE; (ii) the final results reported in [32] use a 20s (COHFACE) and 30s (UBFC, PURE) input time window, while we use a shorter 10s window consistently across datasets. In Table 2 of [32] the effect of window length is shown for COHFACE: they report 1.8 RMSE with window length 400, and 4.7 with 256. Our *self-supervised* baseline achieved 4.6 RMSE with window length 300 (25 run average). This shows that over a range of datasets we can achieve performance comparable to supervised deep learning methods (*e.g.* [32]) using our approach *without annotations*.

A8. Sensitivity to Regularization Parameters

In Table 7 we show the relative performance of a contrastive model trained on the UBFC dataset as the saliency sparsity regularization weight, w_s, and temporal regularization weight, w_t, are varied in the range [0, 0.1, 1, 10]. We find that model performance is not significantly impacted. The saliency map output is most visible when higher values of the sparsity term w_s are used, although we observe that the best parameters can depend on variables such as dataset and resolution.

References

[1] Target heart rates chart. https://www.heart.org/en/healthy-living/fitness/fitness-basics/target-heart-rates. Accessed: 2021-03-17. 2

[2] Guha Balakrishnan, Fredo Durand, and John Guttag. Detecting pulse from head motions in video. In *CVPR*, 2013. 2

[3] Serge Bobbia, Richard Macwan, Yannick Benezeth, Alamin Mansouri, and Julien Dubois. Unsupervised skin tissue segmentation for remote photoplethysmography. *Pattern Recognition Letters*, 124:82–90, 2019. 1, 2, 6, 7

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607. PMLR, 2020. 3

[5] Weixuan Chen and Daniel McDuff. DeepPhys: Video-Based Physiological Measurement Using Convolutional Attention Networks. In *ECCV*, 2018. 2, 3

[6] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Counting out time: Class agnostic video repetition counting in the wild. In *CVPR*, June 2020. 3

[7] RM Fouad, Osama A Omer, and Moustafa H Aly. Optimizing remote photoplethysmography using adaptive skin segmentation for real-time heart rate monitoring. *IEEE Access*, 7:76513–76528, 2019. 2

[8] John Gideon and Simon Stent. Estimating heart rate from unlabelled video. In *ICCV Workshops*, 2021. 8

[9] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 3

[10] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, pages 1735–1742, 2006. 3

[11] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4, 9

[13] Guillaume Heusch, André Anjos, and Sébastien Marcel. A reproducible study on remote heart rate measurement. *arXiv preprint arXiv:1709.00962*, 2017. 6

[14] Ramin Irani, Kamal Nasrollahi, and Thomas B Moeslund. Improved pulse detection from head motions using DCT. In *VISAPP*, volume 3, pages 118–124. IEEE, 2014. 2

[15] Sungjun Kwon, Hyunseok Kim, and Kwang Suk Park. Validation of heart rate extraction using video imaging on a built-in camera system of a smartphone. In *International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2174–2177, 2012. 2

[16] Eugene Lee, Evan Chen, and Chen-Yi Lee. Meta-rPPG: Remote Heart Rate Estimation Using a Transductive Meta-Learner. In *ECCV*, 2020. 2, 3, 5, 7, 8

[17] Xiaobai Li, Jie Chen, Guoying Zhao, and Matti Pietikainen. Remote Heart Rate Measurement from Face Videos under Realistic Situations. In *CVPR*, 2014. 2

[18] Xiaobai Li, Hu Han, Hao Lu, Xuesong Niu, Zitong Yu, Antitza Dantcheva, Guoying Zhao, and Shiguang Shan. The 1st challenge on remote physiological signal sensing (RePSS). In *CVPR Workshops*, June 2020. 6

[19] Ewa Magdalena Nowara, Tim K Marks, Hassan Mansour, and Ashok Veeraraghavan. SparsePPG: Towards driver monitoring us camera-based vital signs estimation in near-infrared. In *CVPR Workshops*, pages 1272–1281, 2018. 6

[20] Daniel J McDuff, Ethan B Blackford, and Justin R Estepp. The impact of video compression on remote cardiac pulse measurement using imaging photoplethysmography. In *IEEE International Conference on Automatic Face & Gesture Recognition*, pages 63–70, 2017. 6

[21] J Moreno, J Ramos-Castro, J Movellan, E Parrado, G Rodas, and L Capdevila. Facial video-based photoplethysmography to detect HRV at rest. *International journal of sports medicine*, 36(06):474–480, 2015. 2

[22] Xuesong Niu, Hu Han, Shiguang Shan, and Xilin Chen. VIPL-HR: A Multi-modal Database for Pulse Estimation from Less-Constrained Face Video. In *ACCV*, 2018. 2, 3, 6, 7

[23] Xuesong Niu, Zitong Yu, Hu Han, Xiaobai Li, Shiguang Shan, and Guoying Zhao. Video-based Remote Physiological Measurement via Cross-verified Feature Disentangling. In *ECCV*, 2020. 2, 3, 7

[24] E. M. Nowara, T. K. Marks, H. Mansour, and A. Veeraraghavan. Near-infrared imaging photoplethysmography during driving. *IEEE Transactions on Intelligent Transportation Systems*, 2020. 6, 7

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *NeurIPS*, pages 8024–8035. 2019. 6

[26] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. Advancements in noncontact, multiparameter physiological measurements using a webcam. *IEEE Transactions on Biomedical Engineering*, 58(1):7–11, 2010. 2

[27] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *Optics express*, 18(10):10762–10774, 2010. 2

[28] Adria Recasens, Petr Kellnhofer, Simon Stent, Wojciech Matusik, and Antonio Torralba. Learning to zoom: a saliency-based sampling layer for neural networks. In *ECCV*, pages 51–66, 2018. 2, 3, 4, 9

[29] Mohammad Soleymani, Jeroen Lichtenauer, Thierry Pun, and Maja Pantic. A multimodal database for affect recognition and implicit tagging. *IEEE Transactions on Affective Computing*, 3(1):42–55, 2011. 6

[30] Radim Špetlík, Vojtěch Franc, Jan Čech, and Jiří Matas. Visual heart rate estimation with convolutional neural network. In *BMVC*, 2018. 3, 6, 7, 9

[31] Ronny Stricker, Steffen Müller, and Horst-Michael Gross. Non-contact video-based pulse rate measurement on a mobile service robot. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, 2014. 6

[32] Yun-Yun Tsou, Yi-An Lee, Chiou-Ting Hsu, and Shang-Hung Chang. Siamese-rPPG network: Remote photoplethysmography signal estimation from face videos. In *Proc. of the 35th Annual ACM Symposium on Applied Computing*, pages 2066–2073, 2020. 11

[33] Sergey Tulyakov, Xavier Alameda-Pineda, Elisa Ricci, Lijun Yin, Jeffrey F Cohn, and Nicu Sebe. Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions. In *CVPR*, 2016. 2, 6

[34] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, and William T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Trans. Graph. (Proceedings SIGGRAPH 2012)*, 31(4), 2012. 2, 3

[35] Zitong Yu, Xiaobai Li, and Guoying Zhao. Remote Photoplethysmograph Signal Measurement from Facial Videos Using Spatio-Temporal Networks. In *BMVC*, 2019. 3, 4, 9

[36] Zitong Yu, Wei Peng, Xiaobai Li, Xiaopeng Hong, and Guoying Zhao. Remote heart rate measurement from highly compressed facial videos: an end-to-end deep learning solution with video enhancement. In *ICCV*, pages 151–160, 2019. 2, 3, 6

[37] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z Li. S3FD: Single shot scale-invariant face detector. In *ICCV*, pages 192–201, 2017. 3