An End-to-end Efficient Framework for Remote Physiological Signal Sensing

Chengyang Hu[1]* Ke-Yue Zhang[2]* Taiping Yao[2] Shouhong Ding[2]†
Jilin Li[2] Feiyue Huang[2] Lizhuang Ma[1]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University,
[2]Youtu Lab, Tencent, Shanghai
{zkyezhang, taipingyao, ericshding, jerolinli, garyhuang}@tencent.com,
huchengyang@sjtu.edu.cn, ma-lz@cs.sjtu.edu.cn

Abstract

Remote photoplethysmography (rPPG) is utilized to estimate the heart activities from videos, which has drawn great interest from both researchers and companies recently. Many existing rPPG deep-learning based approaches focus on measuring the average heart rate (HR) from facial videos, which do not provide enough detailed information for many applications. To recover more detailed rPPG signals for the challenge on Remote Physiological Signal Sensing (RePSS), we propose an end-to-end efficient framework, which measures the average heart rate and estimates corresponding Blood Volume Pulse (BVP) curves simultaneously. For efficiently extracting features containing rPPG information, we adopt the temporal and spatial convolution as Feature Extractor, which alleviates the cost of calculation. Then, BVP Estimation Network estimates the frame-level BVP signal based on the feature maps via a simple 1DCNN. To improve the learning of BVP Estimation Network, we further introduce Heartbeat Measuring Network to predict the video-level HR based on global rPPG information. These two networks facilitate each other via supervising Feature Extractor from different level to promote the accuracy of BVP signal and HR. The proposed method obtains the score 168.08 *(M_{IBI}), winning the third place in this challenge.*

1. Introduction

Electrocardiography (ECG) and Photoplethysmograph (PPG) are the common ways to record heart activities, which are vital for many applications, such as face anti-spoofing [27, 4, 25, 10, 26, 8, 9, 24], face forgery detection [15, 20, 2], *etc.* The traditional methods for obtaining the above signals are mostly measured from skin-contact Electrocardiography (ECG) or Blood Volume Pulse (BVP) sensors, which is not convenient for remote diagnosis. To improve such limitations, remote photoplethysmography (rPPG), which aims to measure heart activity remotely and without contact, as shown in Figure 1, has drawn great interest recently [5, 7, 14, 13, 18, 19, 16].

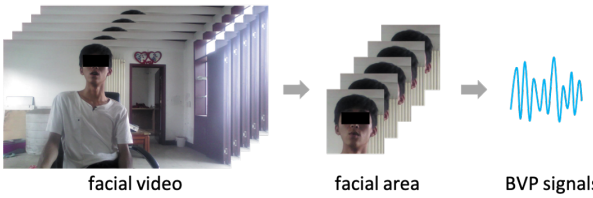


Figure 1. **The illustration of remote photoplethysmography (rPPG).** The rPPG methods aim at measuring heart activity remotely and without contact. First, they analyze the videos and crop the specific facial regions. Second, based on the selected regions, the methods regression the signals.

In the past few years, the researchers put forward several traditional methods to estimate rPPG signals via analyzing the color changes on facial regions [7, 14, 13, 18]. With the rise of deep learning, several methods are proposed to estimate HR based on convolutional neural network (CNN) [11, 17, 3]. While only HR may not provide detailed information for several medical applications, some methods utilize 3DCNN to capture temporal features for rPPG measurement. However, the above methods need large models or many stages to estimate the results, which may cost too many resources in real applications.

To handle this problem, we propose an end-to-end efficient framework to fully utilize the rPPG cues in the facial cheeks for predicting the signals, which estimates the average HR and corresponding Blood Volume Pulse (BVP) curves simultaneously. Specifically, Feature Extract efficiently extracts feature maps containing rPPG information via temporal and spatial convolutions, which processes temporal and spatial information separately to alleviate the cost of computation. Then, BVP Estimation Network utilizes

*Equal Contribution.
†Corresponding Author.

a simple 1DCNN to estimate the frame-level BVP signal based on the feature maps. Further, Heartbeat Measuring Network is introduced to predict the video-level heartbeats, improving the learning of BVP Estimation Network. These last two networks facilitate each other via supervising Feature Extractor from different level to promote the accuracy of BVP signal and HR.

To sum up, the contributions of this work are:

• We propose an end-to-end efficient framework for this RePSS competition, which measures the average heart rate and estimates corresponding Blood Volume Pulse (BVP) curves simultaneously.

• The proposed method obtains the score 168.08 (M_{IBI}), winning the third place in this difficult challenge.

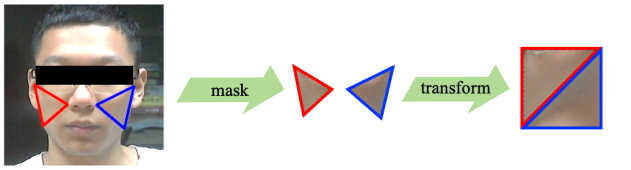
2. Related Work

The research development of rPPG measurement from videos can be split into two stages. Initially, the researchers put forward several traditional methods by inspecting subtle color changes on specific facial regions, including blink source separation [14, 13], least mean square [7], majority voting [6] and self-adaptive matrix completion [18]. However, the traditional methods all rely on mathematical methods, which may not suitable for all situations.

With the rise of deep learning, some methods were proposed for average HR estimation, such as SynRhythm [11], HR-CNN [17] and DeepPyhs [3]. These methods all utilized spatial 2DCNN ignoring the temporal information, which is important for rPPG measurement. To estimate the more detailed rPPG signals, several methods put forward utilize 3DCNN for encoding temporal information. rPPGNet[23] utilized a two-stage framework to enhance the compressed facial videos then estimate the rPPG signals. AutoHR[21] introduced NAS to automatically discover the best-suited backbone for the task of remote HR measurement. RhythmNet [12] trained a CNN-RNN model based on a training set that contains diverse illumination and poses to enhance performance on public dataset. PhysNet [22] constructed both a 3DCNN-based network and an RNN-based network and compared their performance. However, these methods either contain large and complicated models or require many stages to produce the final results, which may cost too many resources in real applications. While our framework is efficient, which is suitable to deploy in real applications.

3. Method

Our method mainly contains three steps. The first step is pre-processing to videos, blood volume pulse (BVP) signals, and heart rate. The second step is BVP signal estimation via an, end-to-end efficient framework. The final step is the post-processing for BVP signals including bandpass filter and peak detection for test dataset prediction which satisfies the requirements of "The 2nd Remote Physiological Signal Sensing (RePSS) Challenge".



(a) Selected ROI (b) Masked ROI (c) Transformed ROI

Figure 2. **ROI selection and transformation.** First, the region of left (red triangle) and right (blue triangle) cheeks are selected and masked. The we transform two triangles into a square as input.

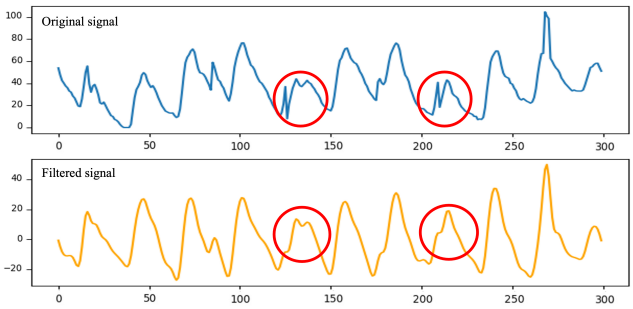


Figure 3. **The original and filtered BVP signals**. The blue and orange line denotes the original and the filtered signal respectively.

3.1. Data Pre-processing

Video Pre-processing. First, we use face detection model to get all face landmarks for each video frame. Since the left and right cheeks have the largest exposed skin, which is more robust in BVP signal estimation, we select these areas as ROI regions based on face landmarks. As shown in Figure 2, triangular ROI of left and right cheeks are converted into a square named Transformed ROI as input for the framework. For a Transformed ROI with the shape $W \times H$, the apexes coordinates of Transformed ROI region are $(0,0),(W-1,0),(0,H-1),(W-1,H-1)$. The transformation of the left cheek is formulated as:

$$\begin{bmatrix} x_1' & y_1' \\ x_2' & y_2' \\ x_3' & y_3' \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ W-1 & 0 \\ 0 & H-1 \end{bmatrix} = A \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{bmatrix}, \quad (1)$$

where $(x_1,y_1),(x_2,y_2),(x_3,y_3)$ are the apexes coordinates of left cheek ROI region, $(x_1',y_1'),(x_2',y_2'),(x_3',y_3')$ are the apexes coordinates of left upper triangle in Transformed ROI, $A \in \mathbb{R}^{3\times3}$ is the transformation matrix. A similar operation is processed to transform the right cheek ROI region to the lower right triangle of Transformed ROI. The final Transformed ROI is the square formed by joining the two transformed regions together as the input of the framework.

BVP Signal Pre-processing. As shown in Figure 3, the original BVP signals contain noise, which leads to the instability of the peak values and the suddenly appearing pulse.

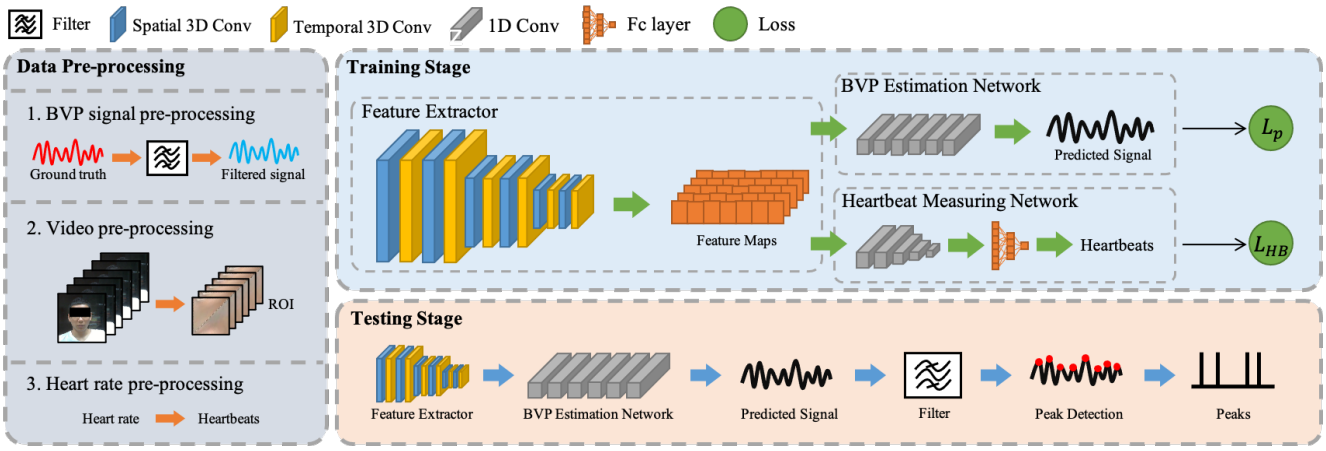


Figure 4. **The overview of our framework.** In training stage, pre-processed BVP signals and heartbeats are utilized for supervisions. First, Feature Extractor obtains the feature maps. Then, BVP Estimation Network and Heartbeat Measurement Network predict rPPG signals and heartbeats respectively. In testing stage, we only use BVP Estimation Network to estimate the signals and detect the peak for the challenge.

To alleviate the difficulty of training, we process BVP signals by a bandpass finite impulse response (FIR) filter with cutoff frequency range [0.5Hz, 5Hz] covering human heart rate range [0.5Hz, 2.5Hz], which not only keeps second harmonic wave and signal information in original signals but also removes noise and smoothes the curve. For facilitating the efficiency of training, we further normalize filtered signals. Finally, we compare the original and filtered BVP signals to demonstrate the differences in Figure 3.

Heart Rate Pre-processing. Directly utilizing heart rate as supervision is meaningless, because the frame rate of each video is not the same, which causes each fixed-length input to represent different time intervals. To handle this issue, we use the number of the heartbeats as supervision, which is formulated as:

$$HB = HR \cdot \frac{f_{input}}{FR}, \tag{2}$$

where HB is the heatbeats, HR is the heart rate, f_{input} is the length of the frames, FR is the frame rate of the video.

3.2. Our Framework

To efficiently predict BVP signals, we propose an end-to-end efficient framework inspired by [1]. The overview of our framework is presented in Figure 4. First, Feature Extractor utilizes the temporal and spatial convolutions to extract the rPPG related feature maps from the Transformed ROI. Then, BVP Estimation Network is composed by 1D convolutions to predict BVP signals at frame-level based on the feature maps. Heartbeat Measuring Network has a familiar 1DCNN structure with BVP Estimation Network. We compress the information in temporal dimension to regress the heartbeat times.

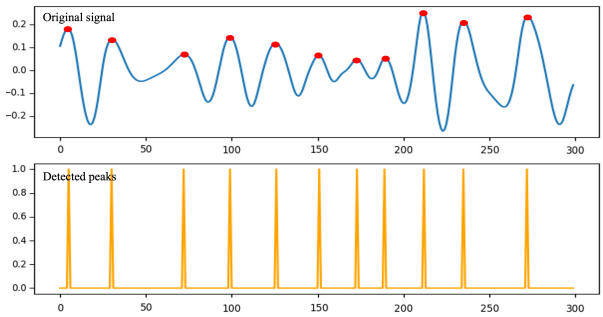


Figure 5. **Peak detection.** The blue line is the predicted signal. The orange line shows the position of the peak. Ones indicate the peak frames of the signals and zeros indicate non-peak frames.

3.2.1 Feature Extractor

Feature Extractor contains several temporal and spatial convolutions to alleviate the cost of computation. Specifically, on one hand, the convolution on the spatial scale is used to obtain the attention of different frames and to combine the features of different color spaces with a nonlinear method. On the other hand, the convolution on the time scale extracts temporal features. Unlike the common 3D convolution with kernel size $t \times w \times h$, our 3D convolution structure has a better performance in signal estimation and has fewer parameters. The detailed structure is presented in Table 1.

3.2.2 BVP Estimation Network

BVP Estimation Network predicts BVP signals based on extracted features, which is a 1-dimension signal with 32 channels. We utilize 1DCNN sequence to analyze the signals at frame-level, which filters the signals and combines channels with a nonlinear function. In the last layer, Sigmoid activation is utilized to map the magnitude of predicted signals into the range $(0, 1)$. The structure of BVP estimation network is described in Table 1.

Table 1. **The detailed structure of our framework.** Feature Extractor contains multiple 3DCNN layers to extract spatial-temporal features efficiently. BVP Estimation Network and Heartbeat Measuring Network contain 1DCNN to predict BVP signals and heartbeat respectively.

Feature Extractor					BVP Estimation Network			Heartbeat Measuring Network		
Layer	Channel	Kernel Size	Stride	Output Size	Layer	Channel	Kernel Size	Layer	Channel	Kernel Size
Input size: $100 \times 128 \times 128$, Input channel: 3					Input size: 100, Input channel: 32			Input size: 100, Input channel: 32		
3DConv1-1	16	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 128 \times 128$	1DCNN1-1	16	3	1DCNN2-1	16	3
3DConv1-2	16	$3 \times 1 \times 1$	$1 \times 1 \times 1$	$100 \times 128 \times 128$	1DCNN1-2	16	3	1DCNN2-2	16	3
3DConv1-3	16	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 128 \times 128$				Max_pool2-1		2
3DConv1-4	16	$3 \times 1 \times 1$	$1 \times 1 \times 1$	$100 \times 128 \times 128$	1DCNN1-3	32	3	1DCNN2-3	32	3
Avg_pool1-1			$1 \times 2 \times 2$	$100 \times 64 \times 64$				Max_pool2-1		2
3DConv2-1	32	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 64 \times 64$	1DCNN1-4	64	3	1DCNN2-4	64	3
3DConv2-2	32	$5 \times 1 \times 1$	$1 \times 1 \times 1$	$100 \times 64 \times 64$				Max_pool2-1		2
3DConv2-3	32	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 64 \times 64$	1DCNN1-5	128	3	1DCNN2-5	128	3
3DConv2-4	32	$5 \times 5 \times 5$	$1 \times 1 \times 1$	$100 \times 64 \times 64$				Global_pool2-1		2
Avg_pool1-2			$1 \times 2 \times 2$	$100 \times 32 \times 32$						
3DConv3-1	32	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 32 \times 32$	1DCNN1-6	1	3	Linear2-1	30	
3DConv3-2	32	$5 \times 1 \times 1$	$1 \times 1 \times 1$	$100 \times 32 \times 32$				Linear2-2	30	
3DConv3-3	32	$1 \times 3 \times 3$	$1 \times 1 \times 1$	$100 \times 32 \times 32$				Linear2-3	1	
3DConv3-4	32	$5 \times 1 \times 1$	$1 \times 1 \times 1$	$100 \times 32 \times 32$						
Global_pool1-1				$100 \times 1 \times 1$						

3.2.3 Heartbeat Measuring Network

Heartbeat Measuring Network predicts heartbeat times of each input clip of videos, which contains a similar 1DCNN with BVP Estimation Network. Since Heartbeat Measuring Network predicts a value but not a signal, we use pooling to compress information along the time dimension. Then the linear function is utilized to calculate the number of the heartbeats at video-level. The structure of Heartbeat Measuring Network is presented in Table 1.

3.2.4 Loss Function

Negative Pearson Correlation. Since Negative Pearson Correlation is widely used in previous works [23, 21] to indicate the similarity of two signals, we strictly follow them to introduce it for supervising on predicted signal results. Pearson Correlation shows the linear similarity between two signals, which only cares about the trend of signals but ignores the magnitude and phase. Specifically, it is formulated as:

$$L_p = 1 - \frac{Cov(s_{pre}, s_{gt})}{\sqrt{Cov(s_{pre}, s_{pre})}\sqrt{Cov(s_{gt}, s_{gt})}}, \quad (3)$$

where s_{pre} and s_{gt} are the predicted signals and ground truth of filtered BVP signal, $Cov(x, y)$ is the covariance of x and y. The Pearson Correlation coefficient measures the strength of the linear relationship between two variables, which takes values in the closed interval $[-1, 1]$. The value 1 reflects a perfect positive correlation between two variables, whereas the value 0 indicates that no correlation can be found. The value -1 reflects a perfect negative correlation between two variables.

Mean Square Error. Mean square error shows the difference between the predicted heartbeats and the ground truth. For heart rate prediction, we use mean square error as loss function which is formulated as:

$$L_{HB} = \frac{1}{K}\sum_{k=0}^{K}(HB_{pre} - HB_{gt})^2, \quad (4)$$

where HB_{pre} and HB_{gt} are predicted heartbeats and ground truth of heartbeats. K is the number of the samples.

Training Loss. The overall loss function is summarized as:

$$L_{all} = L_p + \lambda L_{HB}, \quad (5)$$

where $\lambda \in [0, 1]$ is the weight for balancing the loss.

3.3. Peak Detection

The final submission result of the competition is a binary peak signal with only zeros and ones. The predicted signals need to filter and detect the peak to meet the requirements.

Prediction Filter. Predicted signals contains noise which degrades the performance, so we filter them in a small range of frequency to get more specific signals.

$$s_{filt} = \text{bandpass\_filter}(s_{pre}, low, high), \quad (6)$$

where s_{filt} is the filtered signal, s_{pre} is the original predicted signal, low and $high$ indicate the frequency range.

Peak Detection. The peaks are detected with peak detection algorithm, where the first order difference of signal is:

$$\Delta s = \frac{ds}{dt} = s(t+1) - s(t), \quad (7)$$

where $t = 0, 1, \ldots, T-1$, T is the length of the frames. The collection of peak indexes P satisfies,

$$\Delta s(t+1) < 0, \Delta s(t) > 0, t \in P \quad (8)$$

4. Experiment Result

4.1. Datasets

The challenge provides two datasets: VIPL-HR dataset and OBF dataset. At the training stage, we only use VIPL-

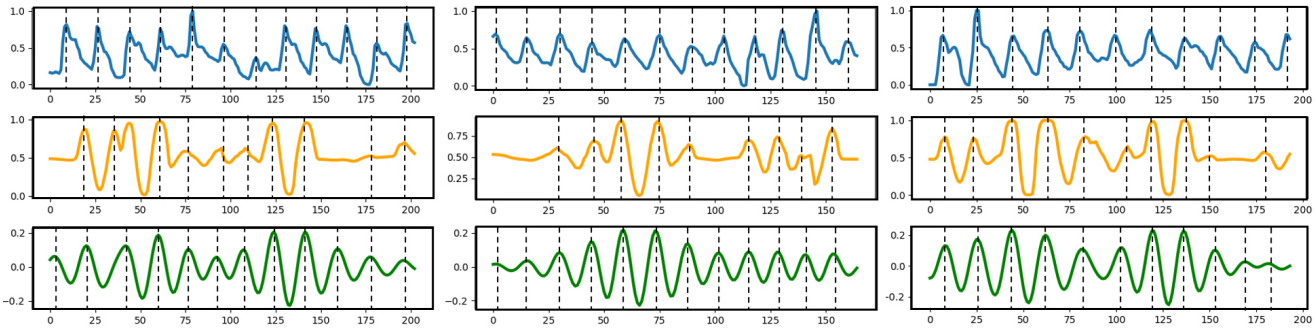


Figure 6. **Some predicted results on validation datasets.** Blue line is the ground truth BVP signal. Orange line is original predicted signals and green line is filtered signal. Dotted line shows the position of the peaks.

HR dataset. At the testing stage, we evaluate the performance of our method on VIPL-HR dataset and OBF dataset.

VIPL-HR Dataset. It provides 2500 videos (500 subjects) as training dataset and 500 videos (100 subjects) as testing dataset. To evaluate the performance of our method offline, we randomly select 500 videos from the training set as the validation set. The videos in this dataset are recorded in complex situations including illumination, movement, age and skin color of subjects, which remains a great challenge for training. 1) Illumination. There are mainly three kinds of illumination. The first type is in the natural environment without artificial light sources, the second one is in the natural environment with natural light sources. The third kind is in the laboratory environment with an adequate light source. 2) Movement. All movements can be divided into three kinds: stable, large motion and talking. The maximum rotation amplitudes are 92° in roll, 105° in pitch and 104° in yaw. And some samples may have sudden, large head twists. 3) Subjects. The subjects include all ages from preschoolers to the elderly. Although the subjects are all of the Asian ethnicity, their skin colors are different, from bright to dull. The heart rate range from 47 bpm to 146 bpm covers the typical heart rate range. All these elements contribute to a great challenge to learn generalized features.

OBF Dataset. This dataset is obtained under tightly controlled conditions, where illumination is controlled and all subjects are stable. All subjects sit in front of the camera with two LED lights illuminating the faces at a 45-degree angle with a distance of 1.5 meters. OBF dataset has diverse subjects with a large age range from 16 to 68, and multiple ethnics including Caucasian, Asian and others. OBF dataset is only used for testing as a domain shift dataset, which includes 500 videos (100 subjects) as the test dataset.

4.2. Evaluation Metrics

Following the competition, we utilize five metrics to evaluate the performance of our framework on test and validation datasets, mean of IBI error (M_{IBI}), standard deviation of IBI error (SD_{IBI}), mean absolute error of heart rate (MAE_{HR}), root mean squared error of heart rate ($RMSE_{HR}$) and Pearson correlation coefficient of heart rate (R_{HR}), which are all explained in detail as follows.

$\boldsymbol{M_{IBI}}$ calculates the mean difference between the predicted IBI curve $R_1(t)$ and ground truth IBI curve $R_2(t)$.

$$AE = \sum_{i=0}^{T} |R_1(t) - R_2(t)|, M_{IBI} = \frac{1}{K}\sum_{k=0}^{K} AE_k, \quad (9)$$

where T is length of frames and K is number of videos.

$\boldsymbol{SD_{IBI}}$ indicates the degree of dispersion of the errors.

$$SD_{IBI} = \sqrt{\frac{1}{K}\sum_{k=0}^{K}(AE_k - M_{IBI})^2}, \quad (10)$$

where AE_k is the absolute error of k-th video and M_{IBI} is mean of IBI error.

$\boldsymbol{MAE_{HR}}$ shows the mean difference between heart rate of prediction and ground truth.

$$MAE_{HR} = \frac{1}{K}\sum_{k=0}^{K}|HR_1^k - HR_2^k|, \quad (11)$$

where HR_1, HR_2 are the heart rate of prediction and ground truth and K is the number of the videos.

$\boldsymbol{RMSE_{HR}}$ presents the degree of dispersion between the errors of heart rate.

$$RMSE_{HR} = \sqrt{\frac{1}{K}\sum_{k=0}^{K}(|HR_1^k - HR_2^k| - MAE_{HR})^2}, \quad (12)$$

where MAE_{HR} is the mean absolute error of heart rate.

$\boldsymbol{R_{HR}}$ demonstrates correlation between predicted heart rate and ground truth heart rate.

$$R_{HR} = \frac{Cov(HR_1, HR_2)}{\sqrt{Cov(HR_1, HR_1)}\sqrt{Cov(HR_2, HR_2)}} \quad (13)$$

where HR_1 and HR_2 means the prediction and ground truth of heart rate.

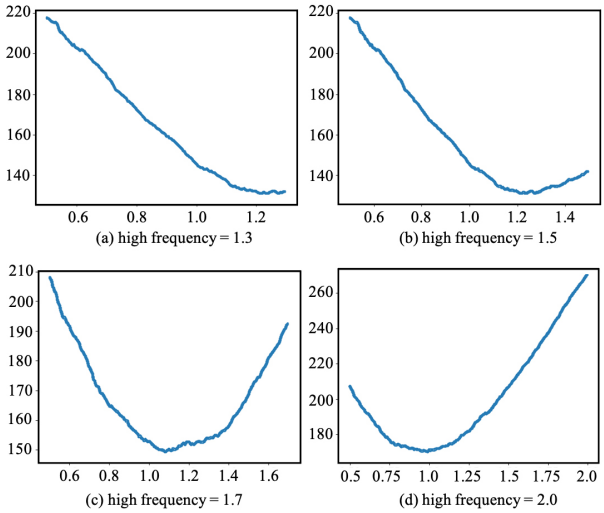


Figure 7. **Relationship between M_{IBI} and frequency range of filter.** The x-axis represents the minimum value of the filter range. The y-axis shows M_{IBI} on validation dataset. When filter frequency ranges in [1.2Hz, 1.3Hz], our framework performs best.

4.3. Implementation Details

The input contains 3 channels in RGB color space with the shape of 128×128. All experiments and network training are done with two GeForce RTX 2080 GPUs using Pytorch framework on Linux platform. We set the batch size, $B = 12$, the length of frames in each input samples, $T = 100$, the adaptive learning rate of Adam optimizer $\beta_1 = 0.9, \beta_2 = 0.999$ and the weight of loss function, $\lambda = 0.1$. We sample video clips every 5 frames, which enriches the data amount from 2500 samples to approximately 50000. The filtering range of the bandpass filter in pre-processing is between 0.5 Hz and 5 Hz, level equals 3.

4.4. Test Results

Results on Validation Dataset. To evaluate the performance of our models, we first test our model on the validation dataset. The metrics mentioned in Section 4.2 are used to evaluate the performance, as shown in Table 2. Our method attains pretty low M_{IBI} 120.43 on validation sets, which proves the effectiveness of the method.

Results on Test Dataset. We test the model, which performs best on the validation sets, on the VIPL-HR datasets and OBF datasets to get final results, as shown in Table 2. Compared with the results on validation dataset, the performance drops a little since the testing sets containing the cross-domain videos. In the challenge, we got third place.

4.5. Ablation Study

frequency range of the bandpass filter is an important component in signal analysis and processing. Filter makes the signals smoother and removes the noise from signals. The frequency range of filter is an important parameter and influence the performance a lot. Figure 7 shows the relationship between M_{IBI} and frequency range of filter. It indicates that when the filter frequency ranges from 1.2Hz to 1.3Hz, our framework has the best performance. In Table 2, we show some specific results with different frequency range on validation dateset. The distribution of the heart rate values conformed to a normal distribution and the mean value is about 75 beats per minute (1.25Hz). Therefore, the selected frequency range, although relatively narrow, can cover the heart rate range of most people.

Table 2. **Results on validation dataset and test dataset.**

	Freq. Range		Results				
	low	high	M_{IBI}	SD_{IBI}	MAE_{HR}	$RMSE_{HR}$	R_{HR}
Valid	0.5	5	235.62	94.54	28.36	32.14	0.003
	0.5	2.5	204.49	78.63	17.47	20.87	0.03
	1	1.5	144.61	59.00	11.22	13.75	0.08
	1.2	1.3	120.43	63.95	10.97	13.24	0.06
Test	1.2	1.3	168.08	162.82	11.84	14.51	0.02

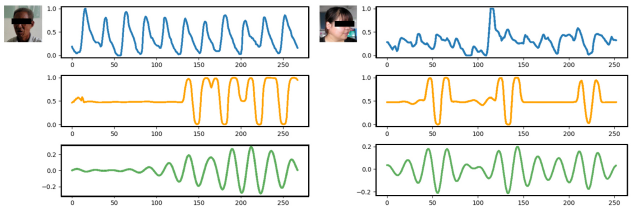


Figure 8. **Some badcase of predicted signals.** Blue line is the ground truth BVP signal. Orange line is original predicted signals and green line is filtered signal. The left picture is one frame from the video.

4.6. Badcase Analysis

In this section, we explore some failure cases in validation datasets to analyze the reason. As shown in Figure 8, the left one is a subject with deep dark skin and shakes head in the last 5 seconds, which induces the model to concentrate more on the last 5 seconds and pay less attention on the deep dark skin. The right subject shakes head suddenly three times, which induces model to predict three obvious peaks in the signal. The results suggest that the deep color of skin and quick motion may influence the results a lot.

5. Conclusion

In this work, we propose an end-to-end efficient framework to estimate BVP signals from processed ROI regions in videos. Feature Extractor is composed of spatial convolutions and temporal convolutions to extract spatial-temporal features. BVP Estimation Network consists of 1D convolutions to estimate BVP signals by extracted features at frame-level. We also introduce Heartbeat Measuring Network which has a similar structure with BVP Estimation Network as auxiliary supervision to predict heartbeat at video-level. We investigate the frequency range of the bandpass filter to detect the range with optimal effect on the validation dataset and utilize this range in the test dataset.

References

[1] M. Artemyev, M. Churikova, M. Grinenko, and O. Perepelkina. Neurodata lab's approach to the challenge on computer vision for physiological measurement. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020.

[2] Shen Chen, Taiping Yao, Yang Chen, Shouhong Ding, Jilin Li, and Rongrong Ji. Local relation learning for face forgery detection. *arXiv preprint arXiv:2105.02577*, 2021.

[3] Weixuan Chen and Daniel McDuff. Deepphys: Video-based physiological measurement using convolutional attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[4] Zhihong Chen, Taiping Yao, Kekai Sheng, Shouhong Ding, Ying Tai, Jilin Li, Feiyue Huang, and Xinyu Jin. Generalizable representation learning for mixture domain face anti-spoofing. *arXiv preprint arXiv:2105.02453*, 2021.

[5] Gerard De Haan and Vincent Jeanne. Robust pulse rate from chrominance-based rppg. *IEEE Transactions on Biomedical Engineering*, 2013.

[6] Antony Lam and Yoshinori Kuno. Robust heart rate measurement from video using select random patches. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[7] Xiaobai Li, Jie Chen, Guoying Zhao, and Matti Pietikainen. Remote heart rate measurement from face videos under realistic situations. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[8] Bofan Lin, Xiaobai Li, Zitong Yu, and Guoying Zhao. Face liveness detection by rppg features and contextual patch-based cnn. In *International Conference on Biometric Engineering and Applications (ICBEA)*, 2019.

[9] Shubao Liu, Ke-Yue Zhang, Taiping Yao, Mingwei Bi, Shouhong Ding, Jilin Li, Feiyue Huang, and Lizhuang Ma. Adaptive normalized representation learning for generalizable face anti-spoofing. *arXiv preprint arXiv:2108.02667*, 2021.

[10] Shubao Liu, Ke-Yue Zhang, Taiping Yao, Kekai Sheng, Shouhong Ding, Ying Tai, Jilin Li, Yuan Xie, and Lizhuang Ma. Dual reweighting domain generalization for face presentation attack detection. *arXiv preprint arXiv:2106.16128*, 2021.

[11] Xuesong Niu, Hu Han, Shiguang Shan, and Xilin Chen. Synrhythm: Learning a deep heart rate estimator from general to specific. In *International Conference on Pattern Recognition (ICPR)*, 2018.

[12] Xuesong Niu, Shiguang Shan, Hu Han, and Xilin Chen. Rhythmnet: End-to-end heart rate estimation from face via spatial-temporal representation. *IEEE Transactions on Image Processing*, 2019.

[13] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. Advancements in noncontact, multiparameter physiological measurements using a webcam. *IEEE Transactions on Biomedical Engineering*, 2010.

[14] 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*, 2010.

[15] Hua Qi, Qing Guo, Felix Juefei-Xu, Xiaofei Xie, Lei Ma, Wei Feng, Yang Liu, and Jianjun Zhao. Deeprhythm: Exposing deepfakes with attentional visual heartbeat rhythms. In *Proceedings of ACM International Conference on Multimedia (ACMMM)*, 2020.

[16] Jingang Shi, Iman Alikhani, Xiaobai Li, Zitong Yu, Tapio Seppänen, and Guoying Zhao. Atrial fibrillation detection from face videos by fusing subtle variations. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[17] Radim Špetlík, Vojtech Franc, and Jirí Matas. Visual heart rate estimation with convolutional neural network. In *Proceedings of the British Machine Vision Conference (BMCV)*, 2018.

[18] 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 *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[19] Wim Verkruysse, Lars O Svaasand, and J Stuart Nelson. Remote plethysmographic imaging using ambient light. *Optics express*, 2008.

[20] Xinyao Wang, Taiping Yao, Shouhong Ding, and Lizhuang Ma. Face manipulation detection via auxiliary supervision. In *International Conference on Neural Information Processing*, 2020.

[21] Zitong Yu, Xiaobai Li, Xuesong Niu, Jingang Shi, and Guoying Zhao. Autohr: A strong end-to-end baseline for remote heart rate measurement with neural searching. *IEEE Signal Processing Letters*, 2020.

[22] Zitong Yu, Xiaobai Li, and Guoying Zhao. Remote photoplethysmograph signal measurement from facial videos using spatio-temporal networks. *arXiv preprint arXiv:1905.02419*, 2019.

[23] 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 *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.

[24] Zitong Yu, Yunxiao Qin, Xiaobai Li, Chenxu Zhao, Zhen Lei, and Guoying Zhao. Deep learning for face anti-spoofing: A survey, 2021.

[25] Jian Zhang, Ying Tai, Taiping Yao, Jia Meng, Shouhong Ding, Chengjie Wang, Jilin Li, Feiyue Huang, and Rongrong Ji. Aurora guard: Reliable face anti-spoofing via mobile lighting system. *arXiv preprint arXiv:2102.00713*, 2021.

[26] Ke-Yue Zhang, Taiping Yao, Jian Zhang, Shice Liu, Bangjie Yin, Shouhong Ding, and Jilin Li. Structure destruction and content combination for face anti-spoofing. *arXiv preprint arXiv:2107.10628*, 2021.

[27] Ke-Yue Zhang, Taiping Yao, Jian Zhang, Ying Tai, Shouhong Ding, Jilin Li, Feiyue Huang, Haichuan Song, and Lizhuang Ma. Face anti-spoofing via disentangled representation learning. *arXiv preprint arXiv:2008.08250*, 2020.