

Architectural Tricks for Deep Learning in Remote Photoplethysmography

Mikhail Kopeliovich Yuriy Mironenko Mikhail Petrushan
Southern Federal University, Center of Neurotechnologies
Rostov-on-Don, Russian Federation

kop@km.ru

Abstract

Architectural improvements are studied for convolutional network performing estimation of heart rate (HR) values on color signal patches. Color signals are time series of color components averaged over facial regions recorded by webcams in two scenarios: Stationary (without motion of a person) and Mixed Motion (different motion patterns of a person). HR estimation problem is addressed as a classification task, where classes correspond to different heart rate values within the admissible range of [40; 125] bpm. Both adding convolutional filtering layers after fully connected layers and involving combined loss function where first component is a cross entropy and second is a squared error between the network output and smoothed one-hot vector, lead to better performance of HR estimation model in Stationary and Mixed Motion scenarios.

1. Introduction

Wide-spreading of cheap color cameras, especially built into smartphones, makes efforts to retrieve biosignals remotely, out of video and without special sensors, very appealing.

Early papers [30, 34, 36] demonstrated this possibility, with information retrieved through analysis of small fluctuations of skin color. This approach was later called iPPG (imaging photoplethysmography) or rPPG (remote photoplethysmography), which are effectively synonyms (possibly, except the specific cases like one-pixel camera [33]). Later papers also demonstrated analysis of micro-movements caused by pulse (imaging ballistocardiography) [2, 24].

More recent works demonstrated that pulse signal can be recovered from long distances [25] up to 50 m [1] and from images as small as 41×30 pixels [14].

Motion of the subject is a significant challenge. The standard approach is to extract heart rate (HR) information from a signal based on small fluctuations of skin color (the green

color component is mostly useful [30, 20]), often on basis of its spectrum [30]. However, frequency of the typical subject's movements (head tilts, for example) often fits within the expected HR range, generating strong false signal [4, 20, 30].

Poh *et al.* [21] tried to resolve this issue by Independent Component Analysis (ICA), which tries to distinguish different sources of the final signal [13, 21].

It was demonstrated, however, that even small motion of a subject during natural interaction with the computer causes significant accuracy decrease (compared to controllable no-movements case), and in-door exercise environment makes ICA almost useless. At the same time, similar methods improved by ML (machine learning) techniques shows much better accuracy [16, 17, 19, 22].

Our work aims at studying impact of certain architectural tricks which could contribute to modern ML-based approaches on rPPG.

In particular, we consider classification-based estimation of HR values by convolutional network followed by two fully connected layers.

Outputs of this network, if normalized, can be treated as relative probabilities distribution (we have single output for the every HR value with constant step). This "pseudo-spectrum" is often noisy. In order to suppress noise-related outliers, some processing or filtering method may be applied to this distribution, for example, smoothing. We replace determined processing by convolutional layers assuming to get optimal filtering procedure during training.

2. Related works

Almost all of recent rPPG papers include ML elements for pre-processing, post-processing or as a main element; often it's neural networks and specifically convolutional neural networks (CNN) [6].

Ground truth signal in most cases retrieved by either electrocardiography (ECG) [3, 17, 28] or contact PPG [20, 31, 38]. ECG is quoted to be more reliable [28], while contact PPG is quoted to be closer to rPPG signal retrieved [38], making training easier. Reproduction of ground truth signal

is often the main area of ML-based training [3].

Some mask is often applied to select the so-called “region of interest” (ROI) – area of the frame image without background pixels and with most informative fluctuations [17, 3, 20]. In most cases, the video of the face is processed.

While modern progress in deep learning techniques is believed to provide powerful tools for the rPPG, straightforward approach is facing difficulties:

Small datasets for training. Most of the available datasets include less than 100 subjects. Often only one type of camera is used.

To compensate a little number of subjects, researchers tend to gather a lot of videos from every single subject, which doesn't seem to resolve the problem, since even with long videos the datasets are still relatively small. To overcome this problem, transfer learning approach is used with original data coming from other domains [6, 5] or even generated out from training on mock signals [18]. Transfer learning capability is also used in [3] to measure quality of the proposed method.

Video compression methods, which tend to preserve details which is visible to the naked eye, and suppress mostly invisible (and therefore meant to be not important) details. Another problem is variable frame-rate [23], often generated by video codecs trying to keep constant bit-rate, which causes jitter on periods between frames.

It was observed that non-compressed video, while impractical, is a much better source for iPPG [28], and iPPG accuracy decreases linearly as compression rate increases [14, 15].

Magnification of small motion and color changes of the skin is used to overcome problems caused by video compression and as a method to increase general sensitivity, [4, 7, 35]. Another quite unusual approach is to use 1-pixel camera which has no problems with the bandwidth and therefore needs no compression [33].

The following methods used to handle the motion of the subject and to overcome the related problems like illumination changes:

- ML-based detection of peaks instead of spectrum analysis [20];
- CNN signal post-processing to extract HR information [28];
- CNN pre-processing which is expected to be stable to small movements [28, 29];
- skin reflection model, which is expected to help with noise caused by observational skin color changes caused by different view angles [3];

- attention model – ROI building procedure which pays special attention to moving pixels of image, also expected to distinguish smaller movements from global rigid motion [12, 3, 37];
- spatio-temporal CNN, which is able to extract temporal-based features out of series of 2D images but, compared to the traditional 3D convolutions, uses significantly less parameters for training [38].

Another sources of inspiration for this paper include:

- increasing of quality of events detection when multi-lead ECG used instead of 1-lead [6];
- motion-compensated pixel-to-pixel pulse extraction sensors used to utilize spatial-redundancy of image [23].

The both researches involved multichannel registration of biosignals, which led to more accurate evaluation of spatio-temporal characteristics of signals. This could be explained by the enhancement of manifestation of the common sources of the registered signals in their cross-correlations.

3. Experimental setup

This section describes the self-collected dataset containing 52 videos recorded on three cameras in different motion scenarios. The preprocessed and ground truth data are publicly available (see Section 4.1).

Three cameras were alternately used for video recording: **Cam₁**: Logitech C920 webcam with 1920×1080 (Width \times Height) pixels and WMV2 video codec.

Cam₂: Microsoft VX800 webcam with 640×480 pixels and WMV3 video codec.

Cam₃: Lenovo B590 laptop integrated webcam with 640×480 pixels and WMV3 video codec.

All video sequences were recorded in RGB (24-bit depth) at 15 frames per second (fps) with 60–80 seconds duration. Each frame contains a person's face.

From 2 to 14 video sequences were recorded for each of 8 healthy participants (7 male, 1 female, aged from 24 to 37, with skin-tones categorized from type-I to type-IV on the Fitzpatrick scale). Distribution of reference HR values is shown in Fig. 1. Each subject signed written consent to take part in the tests, which were performed in compliance with the bioethics regulations; experimental protocols were approved by the bioethics committee of the Southern Federal University.

The distances range from the face to webcam was 0.5–0.7 m. The pixel size of the facial area was from 350×350 pixels to 550×550 when using the Cam₁ and from 150×150 to 250×250 when using cameras Cam₂, Cam₃. Each video sequence was recorded at 15 frames per second in daylight

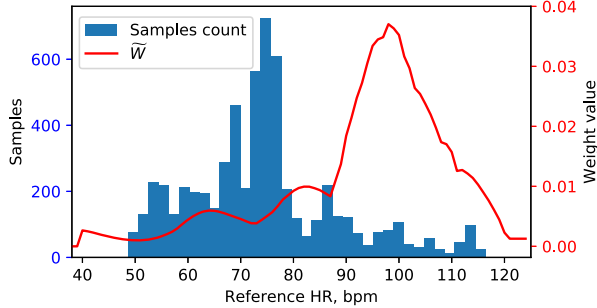


Figure 1. Distribution of HR values in the experimental dataset and corresponding class weights (see Section 4.3).

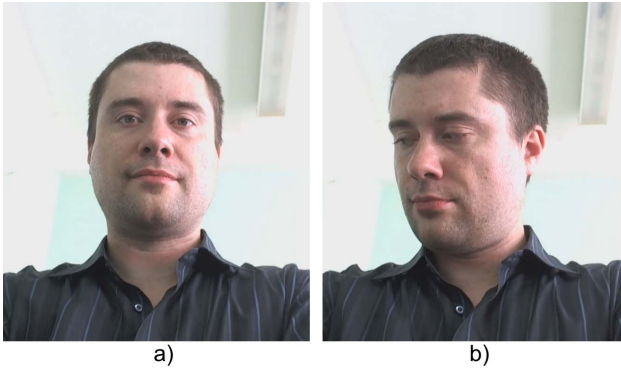


Figure 2. Examples of subject's poses and recording conditions in dataset: Stationary Scenario a) and Mixed Motion Scenario b).

illumination (300–1000 lx). Ground truth, or reference, HR values were obtained by the Choicemmed MD300C318 pulse oximeter (with declared mean absolute error of 2 bpm).

Experiments were conducted in Stationary Scenario and in Mixed Motion Scenario (Fig. 2).

Stationary Scenario. Subject sat still in front of the webcams in a fixed pose looking straight ahead. 12 video sequences for each webcam were recorded.

Mixed Motion Scenario. Subject rotated their head from right to left (with 120° amplitude), from up to down (with 100° amplitude). Subject was asked to speak and change facial expressions. 6 video sequences were recorded for each webcam.

4. Methods

In this section, we describe methods of data pre-processing and HR estimation on a color signal sample by means of CNN model. The final model architecture sequentially performs three steps: feature extraction (convolutional layers in Fig. 3 (a)), HR prediction (fully connected layers in Fig. 3 (b)), and filtering (Fig. 3 (b)).

The following contributions are considered.

Multiple ROIs, forming several input signals (Section (4.1.2)). Our simple attention model focuses on easy-identifiable parts of the face, known to be important [11, 12, 14]. CNN expected to use spatial-redundancy as in [23] and extract cross-correlations between signals like in multi-lead ECG [6].

Pseudo-spectrum instead of regression-like models (Section (4.2)). As long as different sources (blinking, head tilts, mimics) generates noise of different frequencies. Therefore features important to filter them out may differs for different heart rates. Also classification models (compared to regression ones, like used by Spetlik *et al.* [28]) are mentioned to be effective for events detection (HR estimation is believed to be based on detection of a heartbeat events) even if the input is noisy [6].

Combined loss, which is based on the cross-entropy and mean squared error losses (Section (4.3)). In order to minimize the penalty of the minor missclassification (when estimated HR value is close to reference one), we add a mean squared error to the loss function.

Post-processing 1D CNN (Section (4.4)). Magnification of pseudo-spectrum peaks increases contrast and make detection of the HR more accurate. It resembles the post-processing approach of Spetlik *et al.* [28] (but used to increase contrast between classes) and magnification techniques described in [4, 7] (but used for post-processing instead of pre-processing).

4.1. Data pre-processing

The pre-processing is made independently on each given video sequence. It includes extraction of color signals from a sequence and generating of training, validation and test sets. It is assumed that each frame of a video sequence contains face of the same person, while persons can differ in different sequences.

The data containing CS^v and coordinates of ROI_r with synchronized reference HR values are publicly available [10].

4.1.1 Color signal extraction

For a given video sequence, $ROI_r(t)$ is defined as r -th rectangle with coordinates relative to facial bounding box in t -th frame. Facial bounding box is detected by the OpenCV implementation of the Viola–Jones face detector [32] applied to each frame. Regarding the works on ROIs selection [11, 12], six ROIs are used in this paper: $r = 1..6$ (Fig. 4): corresponding to nose, nose bridge, areas under eyes, truncated facial box, and full bounding box. Coordinates of the bounding box are averaged over the last 20 frames (≈ 1.3 sec) to minimize detection jitter.

$CS_r^c(t)$ ($c \in R, G, B$) is a color signal value obtained

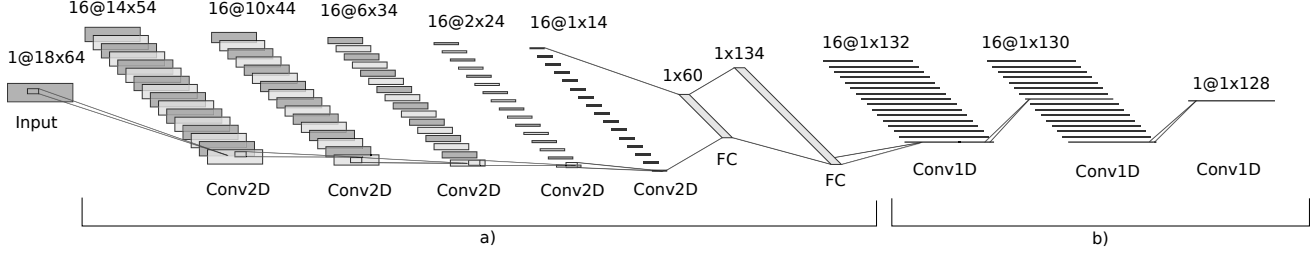


Figure 3. Architecture of CL+F: basic classification network a), and filtering appendix b).

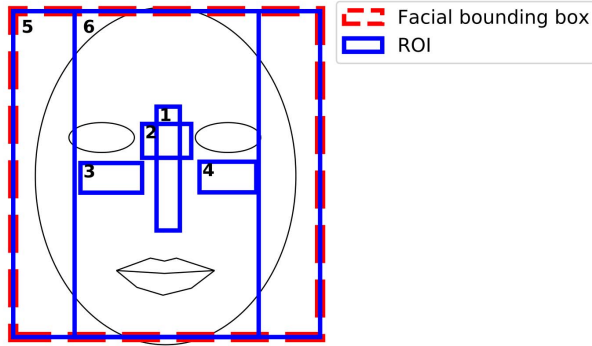


Figure 4. Location of ROIs on the facial bounding box: nose (1), nose bridge (2), areas under eyes (3, 4), truncated facial box (5), full bounding box (6). ROIs coordinates are not ideally symmetrical about a horizontal axis: they were set manually to match the facial features on most video sequences in the dataset.

by averaging intensity of red (R), green (G), or blue (B) color component over the $ROI_r(t)$. Finally, 18 (6×3) one-dimensional color signals $CS_r^c(t)$ were obtained from each video sequence, forming multi-dimensional color signals CS^v , where v is index of video sequence.

4.1.2 Input data generation

The input data samples are 64-frames fragments of color signals CS^v (≈ 4.3 sec per fragment), obtained by splitting each signal into overlapping segments, starting from the first video frame with a step of 10 frames. Samples are scaled to fit the $[-1, 1]$ interval. It is assumed that HR will not change significantly over the sample, so ground truth HR values are averaged within sample resulting in one reference value for each sample.

The video sequences are typically collected in similar environmental conditions but with different participants. In order to keep the training, validation and test sets statistically equivalent, the training set includes first 70% of samples obtained for a CS^v , while the validation set includes next 10% samples (excluding ones that overlap with the training), and test set includes last 20% samples. In this

Camera	Scenario		
	Stationary	Mixed Motion	Both
Cam ₁	1169 / 94 / 340	439 / 27 / 128	1608 / 121 / 468
Cam ₂	944 / 64 / 279	446 / 28 / 132	1390 / 92 / 411
Cam ₃	1056 / 84 / 312	402 / 24 / 120	1458 / 108 / 432
All	3169 / 242 / 931	1287 / 79 / 380	4456 / 321 / 1311

Table 1. Number of samples in the training / validation / test sets per camera and scenario.

way, color signals from each video sequence are presented in all sets, while the training set doesn't intersect with validation or test sets. Alternative distribution of train, validation, and test sets was also evaluated, where the sets were chosen from different non-overlapping video sequences, recorded by different webcams to estimate model generalization properties. Table 1 represents number of samples per each camera and scenario.

The data augmentation was made on the training set, where random uniform noise was added to CS^v values. The noise amplitude was also a uniformly distributed random value with amplitude from $5e-3$ to $5e-2$, that typically corresponds to 5%–50% of the pulse signal amplitude. The amplitude changed after each training step.

4.2. Network architecture

Data sample size is (18×64) , where the 1st dimension is for color signal channels, the 2nd is for discrete time. A sample is processed as a single-channel image. Due to relatively large kernels and, therefore, quick reducing of temporal information through the convolutional layers of the network, we don't use pooling layers to avoid double reducing.

The basic architecture contains five 2D convolution layers with ReLU activation functions followed by two fully connected layers, also with ReLU activations (see Fig. 3(a)). After each convolution layer there is 2D Batch Normalization [8], while after fully connected layers there are 1D Batch Normalization and dropout layers (with 0.5 dropout rate). We tried to add Batch Normalization before and after ReLU, and the latter proved to lead to better accuracy. The number of output channels in convolution layers

is 16. Kernel size is 5×11 (color signal channels \times discrete time) for the first four layers, and 2×11 for 5th layer. Consequently, a 16-channel image of 1×14 size is input to the first fully connected layer, which has 60 output neurons.

We formulate the problem of HR estimation in two ways: as regression or classification tasks. For regression, output of the second fully connected layer is a single value \hat{y} representing HR estimate. For classification, the output $\hat{\mathbf{Y}}$ is a N -length prediction vector, where $N = 128$ is the number of classes. Classes are generated from the range of admissible HR values (40–125 bpm), which is split into N segments of equal size (≈ 0.7 bpm). The segments are assigned to corresponding class labels $\hat{\mathbf{Y}}_i$. The resulting label y is calculated as $\text{argmax}(\hat{\mathbf{Y}}_i)$.

Note that reference value y and estimate \hat{y} correspond to HR values in regression task, while in classification task they are class labels.

4.3. Loss functions

We consider several loss functions: squared error (SE) for regression task; cross entropy (CE) and combined loss (CL) for classification task.

SE loss is calculated between model output \hat{y} and reference HR value y :

$$\text{SE}(\hat{y}, y) = (\hat{y} - y)^2 \quad (1)$$

The distribution of reference HR values in a dataset can be unbalanced. To compensate this, weight coefficients $\tilde{w}_y \in \tilde{\mathbf{W}}$ are involved in all CE losses. First, L vector is calculated denoting inverse numbers of samples per classes in dataset. Next, the weights vector $\tilde{\mathbf{W}}$ is calculated by smoothing the L vector, which is computed as:

$$\tilde{\mathbf{W}} = \frac{L * G^k(\sigma^2)}{\sum_{i=1}^{128} L * G^k(\sigma^2)}, \quad (2)$$

where G^k is zero-mean Gaussian kernel with window size $k = 13$ bpm and $\sigma = \frac{13}{3} \approx 4.3$ bpm. Operator $*$ means discrete convolution with the same padding.

The CE loss combines softmax and negative log likelihood functions:

$$\begin{aligned} \text{CE}(\hat{\mathbf{Y}}, y) &= -\tilde{w}_y \ln \left(\frac{\exp \hat{\mathbf{Y}}_y}{\sum_{i=1}^{128} \exp \hat{\mathbf{Y}}_i} \right) = \\ &= \tilde{w}_y \left(-\hat{\mathbf{Y}}_y + \ln \left(\sum_{i=1}^{128} \exp \hat{\mathbf{Y}}_i \right) \right) \end{aligned} \quad (3)$$

In pure classification task, loss value does not depend on distance between predicted and reference HR values. To take account for the distance, we introduce one-hot vector $\mathbf{Y}^y \in \{0, 1\}^N : Y_i^y = 1 \Leftrightarrow i = y$, corresponding to y label. Similarly to Eq. (2), the vector is smoothed resulting in

$\tilde{\mathbf{Y}}^y$:

$$\tilde{\mathbf{Y}}^y = \frac{\mathbf{Y}^y * G^k(\sigma^2)}{\sum_{i=1}^{128} \mathbf{Y}^y * G^k(\sigma^2)}, \quad (4)$$

where $k = 13$ bpm, $\sigma = \frac{k}{6} \approx 2.2$ bpm.

Finally, combined loss CL is calculated based on CE loss:

$$\text{CL}(\hat{\mathbf{Y}}, y) = \text{CE}(\hat{\mathbf{Y}}, y) + \alpha \cdot \text{MSE}(\hat{\mathbf{Y}}, \tilde{\mathbf{Y}}^y), \quad (5)$$

where $\text{MSE}(\hat{\mathbf{Y}}, \tilde{\mathbf{Y}}^y) = \frac{1}{N} \sum_{i=1}^N (\hat{\mathbf{Y}}_i - \tilde{\mathbf{Y}}_i^y)^2$ is a mean squared error, α is a balancing coefficient equal to 25. The coefficient is selected heuristically in order to equalize the contribution of terms to the sum in Eq. (5) during training.

4.4. Filtering

CL loss function implies a comparison between model prediction $\hat{\mathbf{Y}}$ and smoothed one-hot vector $\tilde{\mathbf{Y}}^y$ (see Eq. (5)). Smoothing parameters σ and k (Eq. (4)) can be hyperparameters, however, their optimization is a challenging problem that should be possibly done after each training iteration. Instead of adding new hyperparameters, we add filtering step to the basic network architecture (Fig. 3(b)), optimizing smoothing of $\hat{\mathbf{Y}}$ during standard training process in order to fit $\tilde{\mathbf{Y}}^y$.

Filtering step goes after the output of second fully connected layer. We use three 1D convolution layers with 16 output channels (single channel for the last layer) and kernel size of 3. Each layer is followed by ReLU activation and 1D Batch Normalization. Due to the layers containing no padding, output of the second fully connected layer increased to 134, so the model output shape remains the same single-channel vector of N length.

We recommend using the filtering step together with CL loss. Nevertheless, it also can be applied when using CE loss in order to clarify a class label. Regarding the regression task, where the model output is a single value, the filtering step is inappropriate.

5. Experimental evaluation and results

This section describes metrics for models evaluation, list of training hyperparameters, and experimental results.

5.1. Evaluation metric

We used two metrics to evaluate the performance of different models of HR estimation. All metrics were applied to results on color signal samples of a test set.

MAE Mean absolute error calculates L1-distance between estimated vector $\hat{\mathbf{Y}}$ of HR values and referent vector \mathbf{Y} :

$$\text{MAE}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{M} \sum_{j=1}^M |\hat{\mathbf{Y}}_j - \mathbf{Y}_j|, \quad (6)$$

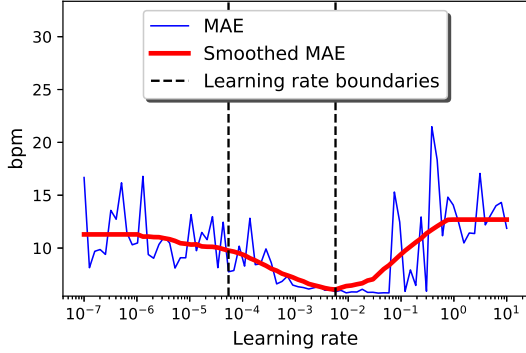


Figure 5. Classification loss as a function of learning rate (for CL+F). The dashed lines indicate the minimum and maximum learning rates ($5.8e-5$ and $5.8e-3$ respectively).

where M is a number of samples. We treat MAE as a qualitative measure of model accuracy.

Coverage at ± 3 bpm This metric was used by Wang *et al.* [33] for analysis of video sequences. We redefine it as a percentage of samples for which MAE value was smaller than 3 bpm. Regarding the classification task, model output is one of 128 classes corresponding to segments within the range of admissible HR values (40–125 bpm). Therefore, for the classification, we use coverage at ± 4 class labels which are approximately equal to ± 2.7 bpm. Coverage metric can be interpreted as model quality. The 3 bpm threshold is close to the MAE of the pulse oximeter (2 bpm) indicating that such a threshold could be used to determine an acceptable measurement.

5.2. Hyperparameters

Here we describe hyperparameters used during the training process. In this work, we set them manually and don't address their optimization.

The batch size was 1024 samples; the number of epochs was 5000. The training set was randomly shuffled after each epoch. The best model parameters were selected from epoch with minimum MAE value on the validation set. The optimization method was Adam [9] with default parameters. The balancing coefficient α for the CL metric (Eq. (5)) is also a hyperparameter.

Learning rate Before the training process, we applied the learning rate range test [26] to choose the learning rate boundaries. The test consisted in the estimation of MAE metric after 5-epochs training for several learning rates varying from $10e-7$ to $10e+1$. The resulting curve (see Fig. 5) was smoothed using Gaussian kernel. The maximum learning rate is defined as argmin of the smoothed curve; minimum learning rate is chosen by dividing the maximum

reduced by two orders of magnitude. During further training, learning rate was linearly changed from minimum to maximum and back according to the “1cycle” learning policy [27].

5.3. Results

We evaluated four models that are titled by the corresponding loss functions: SE model for regression task (62,675 parameters); CE (70,676 parameters), CL (70,676 parameters), and CL+F (with filtering layers, 72,017 parameters) for the classification task.

Training and evaluation methods were implemented in Python (using PyTorch library). The code for generating of the dataset from the color signals as well as the implementation of the proposed architecture, training and testing procedures, and trained models are freely available online [10].

The test set was divided into the several subsets by scenario and used cameras (see Section 3): Stationary and Mixed Motion subsets titled according to the corresponding scenarios, Cam_1 , Cam_2 , Cam_3 subsets containing the samples from the corresponding cameras in both scenarios, and Full test set, which includes all samples of the test set.

5.3.1 Accuracy on test subsets

Model comparison results are presented in Table 2. The considered models were trained on the Full training set (defined in Section 4.1.2). Then the models were evaluated on the test subsets. It is clear that the SE model had much lower accuracy than classification-based models. Accuracy of both CL and CL+F models was higher than of CE, where the distance between classes is not taken into account. Adding filtering layers to the the CL model led to the highest accuracy in most cases including Full test set. The CL model had low coverage value on Mixed Motion and Cam_2 subsets. The former can be explained by the presence of high-amplitude noise in color signals caused by motions.

The coverage metric estimations of the CL+F model were typically near 50%. It is insufficient to use the model in a practical applications. However, the size of the training set was nearly 15 times smaller than the number of the model parameters. Therefore, the model accuracy and coverage could grow with the dataset expansion.

Fig. 6 shows scatter plots for the considered models evaluated on the different test subsets. Predictions of the SE model were distributed within the first half of admissible HR range as the result of the unbalanced dataset (see Fig. 1). Classification-based models led to the similar plots differing in a number of outliers, where the CL+F model showed the best results.

As said before, head motion causes disturbances in the color signal. The amplitude of disturbances is up to two orders higher than amplitude of the signal. Due to that, it had

<i>Test subset</i>	Stationary		Mixed Motion		Cam ₁		Cam ₂		Cam ₃		Full test set	
<i>Model</i>	MAE	Cover	MAE	Cover	MAE	Cover	MAE	Cover	MAE	Cover	MAE	Cover
SE	11.6	10.8%	15.4	13.7%	13.3	13.7%	11.7	12.4%	12.9	9.0%	12.7	11.7%
CE	4.4	44.8%	11.1	31.3%	5.8	36.8%	7.5	38.7%	5.6	47.2%	6.3	40.9%
CL	3.8	48.3%	8.9	39.5%	5.5	43.8%	6.4	34.5%	3.9	58.1%	5.3	45.8%
CL+F	4.1	47.8%	6.9	48.7%	4.6	44.7%	5.6	45.0%	4.5	54.2%	4.9	48.1%

Table 2. Evaluation results of the considered models. The models were trained on the full training set and evaluated on the subsets of the test set. Cover means Coverage at ± 3 bpm metric. The best values are in bold.

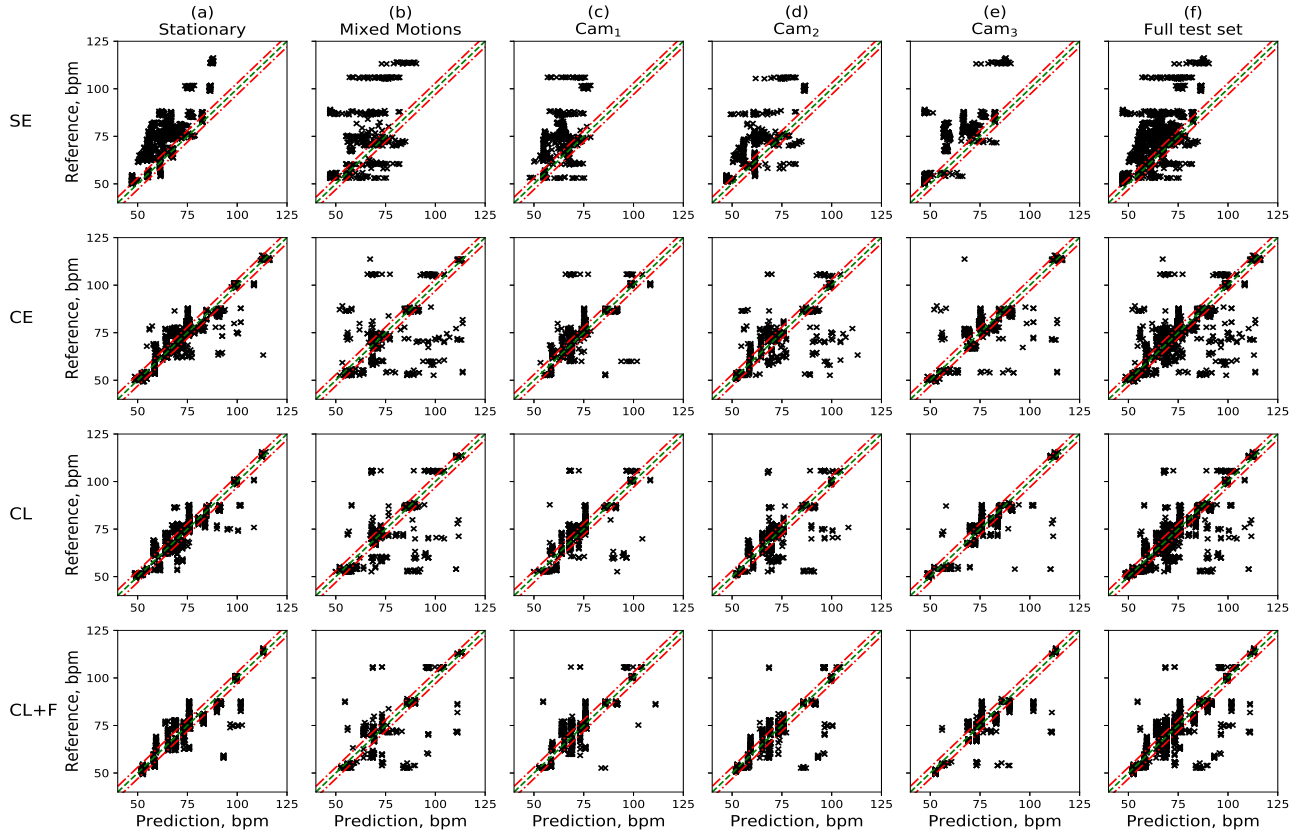


Figure 6. Scatter plots of reference HR values and HR estimations of models trained on the full training set. From top to bottom: models SE, CE, CL, CL+F. From left to right: test subsets a) Stationary, b) Mixed Motion, c) Cam₁ only, d) Cam₂ only, e) Cam₃ only, f) Full test set. Dashed line is a zero-error line, two dash-dotted lines form error interval of 3 bpm.

been unexpected for the model to have the same coverage values on the Stationary and Mixed Motion subsets, which was true for CL+F. Different MAE values indicate a large number of outliers (Fig. 6(b)) on the Mixed Motion subset. We believe that the filtering out of such outliers merits further research.

5.3.2 Model generalization

We studied the generalization of the CL+F network architecture by comparing of the models trained on the different subsets: $(CL+F)_i$ ($i \in \{1, 2, 3\}$) trained on the training sub-

Model	Test subset			
	Cam ₁	Cam ₂	Cam ₃	Full test set
$(CL+F)_1$	4.9	13.3	17.2	11.6
$(CL+F)_2$	20.5	5.0	16.6	14.3
$(CL+F)_3$	18.2	20.3	4.8	14.4
$(CL+F)_{1,2}$	3.1	4.6	14.8	7.5
$(CL+F)_{1,3}$	7.6	13.5	7.1	9.3
$(CL+F)_{2,3}$	10.1	5.5	4.6	6.8

Table 3. MAE values of CL+F trained on different training subsets.

sets with samples from the single camera Cam_i ; $(\text{CL}+\text{F})_{i,j}$ ($i, j \in \{1, 2, 3\}, i \neq j$) trained on the training subsets with the samples from two cameras $\text{Cam}_i, \text{Cam}_j$. Sets based on two cameras were reduced by removing random 50% of the samples in order to equalize a number of samples.

MAE values are given in the Table 3. In the single-camera case, the error was high on the every test subset excluding one corresponding to the camera. This is due to the different camera resolutions, noise, codecs, and other parameters. Two-cameras case led to the similar results: low error for cameras from the training subset and high errors for the remaining camera. However, the error for the remaining camera was noticeably lower than the errors on cameras out of training subset in the single-camera case. Moreover, two-camera cases provided better accuracy on the full test set. As the every training subset had a comparable number of samples, we conclude that the CL+F network architecture provides a high generalizing ability for its instances.

6. Conclusion

The problem of remote photoplethysmography by means of deep learning was considered. Color signals, which are time series of red, green, and blue color components averaged over certain regions in the facial area (cheeks, forehead, nose, etc.), were used as inputs. Inputs were processed by convolutional neural network followed by two fully connected layers. Multiple outputs of this network correspond to different possible HR values, with constant step. The impact of improvements to network architecture and loss function was studied.

In particular, adding convolutional-based filter for post-processing of network outputs led to better accuracy of HR estimations. We expect this improvement can benefit to wide range of deep neural network architectures which address a regression problem by classification and produce “pseudo-spectrum” as output.

Another improvement is the combined loss function, where the first component is a cross entropy and the second one is a mean squared error between the network output and smoothed one-hot vector. The proposed model demonstrated generalization tendency: the model performance, which was evaluated on a particular camera increases with an increasing number of cameras in the training set (excluding the chosen camera); the number of training samples preserved the same.

Acknowledgments. The project is supported by the Russian Ministry for Education and Science, project no.2.955.2017/4.6 “Development of the hardware and software system for monitoring the attention level and psychoemotional state of pilots and dispatching personnel to improve flight safety”.

References

- [1] E. B. Blackford, J. R. Estepp, A. M. Piasecki, M. A. Bowers, and S. L. Klosterman. Long-range non-contact imaging photoplethysmography: cardiac pulse wave sensing at a distance. In *Proc. SPIE 9715, Optical Diagnostics and Sensing XVI: Toward Point-of-Care Diagnostics*, volume 9715, page 971512, 03 2016. 1
- [2] G. Balakrishnan, F. Durand, and J. Guttag. Detecting pulse from head motions in video. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3430–3437, 2013. 1
- [3] W. Chen and D. McDuff. Deepphys: Video-based physiological measurement using convolutional attention networks. In V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, editors, *Computer Vision – ECCV 2018*, pages 356–373. Springer International Publishing, 2018. 1, 2
- [4] W. V. Chen and D. McDuff. Deepmag: Source specific motion magnification using gradient ascent. *ArXiv*, abs/1808.03338, 2018. 1, 2, 3
- [5] O. Faust, U. R. Acharya, and T. Tamura. Formal design methods for reliable computer-aided diagnosis: A review. *IEEE Reviews in Biomedical Engineering*, 5:15–28, 2012. 2
- [6] N. Ganapathy, R. Swaminathan, and T. M. Deserno. Deep learning on 1-d biosignals: a taxonomy-based survey. *Yearbook of Medical Informatics*, 27:098–109, 08 2018. 1, 2, 3
- [7] C. Hurter and D. McDuff. Cardiolens: remote physiological monitoring in a mixed reality environment. In *SIGGRAPH Emerging Technologies*, pages 1–2, 07 2017. 2, 3
- [8] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. 4
- [9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 6
- [10] M. Kopeliovich, M. Petrusan, and Y. Mironenko. rPPG dataset. <https://osf.io/fdrbh/>, 2019. [Online; accessed 22-Aug-2019]. 3, 6
- [11] M. V. Kopeliovich and M. V. Petrusan. Optimal facial areas for webcam-based photoplethysmography. *Pattern Recognition and Image Analysis*, 26(1):150–154, 2016. 3
- [12] M. Kumar, A. Veeraraghavan, and A. Sabharwal. DistancePPG: Robust non-contact vital signs monitoring using a camera. *Biomedical Optics Express*, 6(5):1565, 2015. 2, 3
- [13] R. Macwan, Y. Benezeth, and A. Mansouri. Remote photoplethysmography with constrained ica using periodicity and chrominance constraints. *BioMedical Engineering OnLine*, 17(1):22, Feb 2018. 1
- [14] D. McDuff. Deep super resolution for recovering physiological information from videos. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018. 1, 2, 3
- [15] D. McDuff, E. B. Blackford, and J. Estepp. In *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pages 63–70, 05 2017. 2

- [16] A. V. Moco, S. Stuijk, and G. de Haan. Ballistocardiographic artifacts in ppg imaging. *IEEE Transactions on Biomedical Engineering*, 63:1804–1811, 2016. [1](#)
- [17] H. Monkaresi, R. A. Calvo, and H. Yan. A machine learning approach to improve contactless heart rate monitoring using a webcam. *IEEE Journal of Biomedical and Health Informatics*, 18(4):1153–1160, July 2014. [1](#), [2](#)
- [18] X. Niu, H. Han, S. Shan, and X. Chen. Synrhythm: Learning a deep heart rate estimator from general to specific. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3580–3585, 2018. [2](#)
- [19] E. M. Nowara, T. K. Marks, H. Mansour, and A. Veeraghavan. Sparseppg: Towards driver monitoring using camera-based vital signs estimation in near-infrared. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1353–135309, 2018. [1](#)
- [20] A. Osman, J. Turcot, and R. El Kaliouby. Supervised learning approach to remote heart rate estimation from facial videos. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–6, May 2015. [1](#), [2](#)
- [21] M.-Z. Poh, D. J. McDuff, and R. W. Picard. Non-contact, automated cardiac pulse measurements using video imaging and blind source separation. *OPT EXPRESS*, 18(10):10762–10774, 2010. [1](#)
- [22] S. K. A. Prakash and C. S. Tucker. Bounded kalman filter method for motion-robust, non-contact heart rate estimation. *Biomedical optics express*, 9 2:873–897, 2018. [1](#)
- [23] R. Ribon Fletcher, D. Chamberlain, N. Paggi, and X. Deng. Implementation of smart phone video plethysmography and dependence on lighting parameters. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, volume 2015, pages 3747–3750, 08 2015. [2](#), [3](#)
- [24] D. Shao, F. Tsow, C. Liu, Y. Yang, and N. Tao. Simultaneous monitoring of ballistocardiogram and photoplethysmogram using a camera. *IEEE Transactions on Biomedical Engineering*, 64:1003–1010, 2017. [1](#)
- [25] P. Shi, V. Azorin Peris, A. Echiadis, J. Zheng, Y. Zhu, P. Cheang, S. Cheang, and S. Hu. Non-contact reflection photoplethysmography towards effective human physiological monitoring. *Journal of Medical and Biological Engineering*, 30:161–167, 01 2010. [1](#)
- [26] L. N. Smith. No more pesky learning rate guessing games. *CoRR*, abs/1506.01186, 2015. [6](#)
- [27] L. N. Smith and N. Topin. Super-convergence: Very fast training of residual networks using large learning rates. *CoRR*, abs/1708.07120, 2017. [6](#)
- [28] R. Spetlik, J. Cech, V. Franc, and J. Matas. Visual heart rate estimation with convolutional neural network. In *Proceedings of British Machine Vision Conference*, 2018. [1](#), [2](#), [3](#)
- [29] C. Tang, J. Lu, and J. Liu. Non-contact heart rate monitoring by combining convolutional neural network skin detection and remote photoplethysmography via a low-cost camera. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1390–13906, 2018. [2](#)
- [30] W. Verkruyse, L. O. Svaasand, and J. S. Nelson. Remote plethysmographic imaging using ambient light. *OPT EXPRESS*, 16(26):21434–21445, 2008. [1](#)
- [31] M. Villarroel, J. Jorge, C. Pugh, and L. Tarassenko. Non-contact vital sign monitoring in the clinic. In *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pages 278–285, 05 2017. [1](#)
- [32] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *CVPR*, 1:511–518, 2001. [3](#)
- [33] W. Wang, A. den Brinker, and G. De Haan. Single-Element Remote-PPG. *IEEE Trans Biomed Eng.*, 66(7):2032–2043, 2019. [1](#), [2](#), [6](#)
- [34] F. P. Wieringa, F. Mastik, and A. F. W. Van Der Steen. Contactless multiple wavelength photoplethysmographic imaging: A first step toward “spO 2 camera” technology. *Annals of Biomedical Engineering*, 33(8):1034–1041, 2005. [1](#)
- [35] H.-Y. Wu, M. Rubinstein, E. Shih, J. V. Guttag, F. Durand, and W. T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics - TOG*, 31, 07 2012. [2](#)
- [36] T. Wu. Ppgi: New development in noninvasive and contactless diagnosis of dermal perfusion using near infrared light. *Journal of the Society of Chinese Physists in Germany*, 7(1):17–24, 10 2003. [1](#)
- [37] Z. Yang, X. Yang, and Y. Pang. Motion-tolerant heart rate estimation from face videos using derivative filter. *Multimedia Tools and Applications*, pages 1–11, 2019. [2](#)
- [38] Z. Yu, X. Li, and G. Zhao. Recovering remote photoplethysmograph signal from facial videos using spatio-temporal convolutional networks. *ArXiv*, abs/1905.02419, 2019. [1](#), [2](#)