HeartTrack: Convolutional neural network for remote video-based heart rate monitoring

Olga Perepelkina[1], Mikhail Artemyev[1], Marina Churikova[1,2], and Mikhail Grinenko[1]

[1]Neurodata Lab LLC, Miami, USA
[2]Lomonosov Moscow State University, Faculty of Biology, Department of Higher Nervous Activity, Moscow, Russia
o.perepelkina@neurodatalab.com, m.artemyev@neurodatalab.com, m.churikova@neurodatalab.com, m.grinenko@neurodatalab.com

Abstract

Detection and continuous monitoring of heart rate can help us identify clinical relevance of some cardiac symptoms. Over the last decade, a lot of attention has been paid to the development of the algorigthms for remote photoplethysmography (rPPG). As a result, we can now accurately monitor heart rate of still sitting subjects using data extracted from video feed. Aside from methods based on hand-crafted features, there have also been developed the more advanced learning-based rPPG algorithms. Deep learning methods usually require large amounts of data for training, however, biomedical data often suffers from lack of real-life data. To address these issues, we have developed a HeartTrack convolutional neural network for remote video-based heart rate tracking. This learning-based method has been trained on synthetic data to accurately estimate heart rate in different conditions. Moreover, here we provide two new rPPG datasets - MoLi-ppg-1 and MoLi-ppg-2 - that were recorded in complicated conditions that were close to the natural ones. The datasets include videos that feature moving and talking subjects, different types of lighting, various equipment, etc. We have used our new MoLi-ppg-1 and MoLi-ppg-2 datasets for algorithm training and testing, and the existing UBFC-RPPG dataset for the algorithm testing and comparison with other approaches. Our HeartTrack neural network shows state-of-the-art results on the UBFC-RPPG database (MAE=2.412, RMSE=3.368, R=0.983).

1. Introduction

Heart rate is an important physiological signal that reflects the physical state of a person. This parameter is monitored in the vast majority of healthcare applications. A normal heart rate is usually estimated as 50 to 100 beats per minute [1]. Photoplethysmography (PPG) is a common way of measuring heart activities which is widely used in medicine, sports, and healthcare applications. PPG is an optical method used to measure the light reflected from the skin or variations in transmission intensity. Commonly used photoplethysmography devices contact with a subject's skin which may cause discomfort and be inconvenient in some cases [2]. Remote video photoplethysmography (rPPG) requires only ambient light and a digital camera to capture a person's vital signs. This technique measures heart activity without any physical contact. In recent years, there has been emerging a growing number of studies dedicated to remote pulse rate estimation based on data extracted from face videos [3, 4, 5, 6, 7, 8, 9, 10, 1]. This technology has many potential applications such as remote patient monitoring, neonatal intensive care unit monitoring, driver status assessment, affective state assessment, vivo detection, and etc [3].

Most rPPG algorithms are based on handcrafted features. These approaches often include complex multi-stage methods that are difficult to adjust and implement. Most of the methods require face tracking and registration, skin segmentation, color space transformation, signal decomposition and filtering steps. Other problems of these algorithms are a decreased signal-to-noise ratio for the dark skin [11, 12] and the age changes of skin. Skin of an elderly person is typically thinner, paler, and has wrinkles. Also, number of melanocytes (pigment-containing cells) is decreased, which in turn changes optical features of skin [13]. Aside from methods based on handcrafted features, there are also other learning-based methods designed specifically for remote heart rate estimation. The latter can potentially solve the problems associated with the former ones. Deep learning

has been successfully used in many tasks related to computer vision, especially when large amount of labelled data is available.

We present a novel convolutional neural network (CNN) that learns to detect pulse signal from videos. The proposed method is based on CNN and is capable to learn on new data; moreover, it can be potentially used in a wide range of conditions. The performance of the proposed method can be improved by the increasing of the training set size, in comparison with previous methods. Thus, the proposed method can be used for pulse rate estimation from video signal in any natural conditions and for a person of any age, gender, and skin features.

Our paper has the following novelty: 1. We provide new approach (HeartTrack) that uses synthetic data to pretrain the 1D convolutional part of the CNN, and attention mechanism for the rPPG analysis. 2. HeartTrack shows state-of-the-art results on the UBFC-RPPG database (MAE = 2.412, RMSE = 3.368, R = 0.983). 3. We provide new MoLi-ppg-1 and MoLi-ppg-2 rPPG datasets that: 1) are open for research community, 2) contain complicated close to natural conditions (movements, speech, different lighting, various equipment, etc). 4. We provide baseline solution for MoLi-ppg-1 and MoLi-ppg-2 databases received with described Heart-Track network.

2. Related works

2.1. Remote photoplethysmography methods based on hand-crafted features

A number of denoising methods have been proposed to conduct remote photoplethysmography. One of the main source of noise is movements and changes in lighting. Traditional denoising methods are based on handcrafted features and contain two types of approaches - adaptive region of interest (ROI) selection that aim to obtain the noiseless patch, and color signal processing that aim to separate vital signs from noise. The signal processing methods include typical blind source separation approaches: Independent Component Analysis (ICA) and Principal Component Analysis (PCA). ICA decomposes an RGB signal into components based on the assumption that the input signals corresponding to different sources are statistically independent [14], while PCA maximizes the variance of original points' projection onto components, whereby the source signals are assumed to be uncorrelated [15]. Another group of approaches incorporates a set of model-based methods that rely upon the knowledge of different components' color vectors in the demixing procedure. The group of these methods contains the chrominance model (CHROM), blood volume pulse signature (PBV) model, and a plane orthogonal to the skin (POS) model [3].

Several approaches include different modifications of ROI selection [4, 5, 6, 7]. For example, Kumar et al. [4] presented a method that combined skin-color change signals from a number of patches of the face by using a weighted average with weights depending on blood perfusion and incident light intensity in the patches. Tulyakov et al. [5] proposed a self-adaptive matrix completion approach which dynamically selected the most relevant face ROI for robust pulse estimation. The main drawback of this algorithm is a large number of hyperparameters that need to be tuned. Liu et al. [6] used self-adaptive signal separation to distinguish the noiseless block of facial region with a weight-based scheme. This noiseless signal containing vital information was used to obtain the holistic pulse signal, based on which the average pulse was computed by the means of wavelet transform and data filter. The proposed method was shown to outperform the methods of Kumar et al. and Tulyakov et al. in real-life conditions [6]. Finally, Yang et al. [7] suggested a novel method similar to the one described above, which presents a patch-based fusion framework for accurate pulse estimation in moving subjects.

Thus, we have described two main groups of traditional denoising algorithms that do not require training. Another category of rPPG methods are based on deep-learning models.

2.2. Learning-based rPPG methods

More recently, a few such methods have been proposed for pulse estimation. These methods include SynRhythm [8], HR-CNN [9], DeepPhys [10], and 3D CNN for remote pulse rate measurement [16]. DeepPhys by Chen and McDuff was the first end-to-end system for video-based measurement of pulse using a deep convolutional network [10]. Radim et al. [9] proposed the HR-CNN, which remotely predicts pulse with a two-step convolutional neural network (CNN) using the aligned face images. Another problem is the formation of a training sample due to the lack of real-life data. Often the amount of biomedical data has many "gaps" in the distribution since we cannot, for various reasons, get all of the possible signal options. It is believed that a large scale of training data is needed in order to train a robust neural network and improve its accuracy [17]. For this purpose Niu et al. designed a strategy to train a deep heart rate estimator from a large volume of synthetic PPG signals and a limited number of available face video data. The results of this experiment performed using the public databases show the effectiveness of this approach [8].

Recently Bousefsaf et al. [16] have also proposed a 3D CNN, and a particular training procedure that employs only synthetic data. Authors used a public dataset UBFC-RPPG [18] to demonstrate that this network can effectively extract pulse rate from video without the need for any processing of frames.

3. Our method

3.1. Heart rate estimation pipeline

First, our method detects faces using a RetinaNet network [19] with MobileNet backbone [20] trained with focal loss [19]. The detected regions of interest (ROI) associated with faces are processed independently. We assume that there is only one person in the video in order to simplify the following description. Affine face alignment based on facial landmarks detection [21] is performed for each face. We use ROI average pooling to resize facial areas to the size of $W \times H$ for the heart rate estimation network, where $W = H = 36$. Bandpath filter for [45bpm, 180bpm] frequencies is applied for each (pixel, channel) pair independently in order to filter out signals not related to pulse cycles.

The neural network for heart rate estimation named HeartTrack (see Figure 1 (a)) is described below. It can be trained to evaluate the median heart rate in 8 seconds ($T = 200$ frames) interval in end-to-end manner. A common way to obtain a photoplethysmography signal using the given ROI is a combination of global spatial average pooling and signal source separation methods. While global pooling is an efficient way of getting rid of noise if a face moves or ROI is covered by a foreign object (such as hair or hands), it can refract the signal; such refraction may be difficult to filter out during the next steps. Therefore we use 3D spatio-temporal attention neural network (see Figure 1 (b)) prior to the global pooling. We shall call this network 3D CNN. This network enables us to do three things simultaneously: to choose the ROI that fits best for pulse detection in each frame, to select the optimal nonlinear function of color channels, and to complete signal filtering using temporal information.

The 3D CNN ends with a global spatial pooling layer. Its output has a shape of $batch\_size \times T \times C$, where $C = 32$ is the channels number of the last convolutional layer of the 3D CNN. This way, we have received C time series for a video fragment; each of them having the length of T that can be used as rPPG signals. If the denoising process goes successfully, these time series are supposed to be close to periodical with a period equal to the heart rate of a subject featuring in the video. To identify the main frequency in these time series, we use 1D convolutional neural networks with shared weights. As a result, we are going to get C estimations of a subject's heart rate. We use a feed-forward neural network with one hidden layer with 30 neurons for averaging the outputs of the 1D networks.

We use our MoLi-ppg-1 dataset (30 subjects, 8 hours of data) for model training and validate it on our MoLi-ppg-2 dataset (15 subjects, 3,5 hours of data), and vise versa. Subjects and settings were different in these two datasets to avoid model overfitting. Detaied description of our new datasets can be found in Section 4.1. Additionally, we tested these two models on public UBFC-RPPG dataset [18].

3.2. 3D spatio-temporal attention neural network

3D CNN (see Figure 1 (b)) has 3 inputs: *diff*, *mask* and *frames*.

- *diff* input is a time-domain discrete derivative of the video in ROI. Its size is $batch\_size \times 200 \times 36 \times 36 \times 3$. We use *diff* as the main source of pulse information in our network.
- *mask* is a tensor of size $batch\_size \times T \times W \times H \times 1$. We first define a facial mask where the value of each pixel is equal to 0 if the corresponding pixel in the ROI belongs to eyes or mouth or does not belong to the facial area; otherwise, its value is considered equal to 1. In order to evaluate *mask* tensor, we apply ROI mean pooling to the facial mask. Facial landmarks detection is used for face, mouth and eyes areas localization. *mask* tensor is used in hard attention mechanism to prevent the network from using irrelevant background information from the video.
- *frames* tensor of size $batch\_size \times 200 \times 36 \times 36 \times 3$ consists of video frames content located in the ROI area. It is used in the soft attention mechanism. For example, it can help the model to filter out the areas of the face covered by hair, areas with face paint, or moving face parts.

Diff input goes through two 3D convolutional blocks with subsequent average pooling layers. The first block has 16 channels kernel size = $3 \times 3 \times 3$, and ends with an average pooling layer with kernel size and stride = $1 \times 2 \times 2$;, the second one has 32 channels kernel size = $5 \times 3 \times 3$ and ends with a global average pooling layer.

Each convolutional block ("3D Conv Block" at figure 1 (b)) with kernel size $t \times w \times h$ and c channels has sequential structure and consists of the following layers:

· 3D Convolution, c channels, $kernel\_size = 1 \times w \times h$
· ReLU activation
· 3D Convolution, c channels, $kernel\_size = t \times 1 \times 1$
· Batch Normalization
· ReLU activation
· 3D Convolution, c channels, $kernel\_size = 1 \times w \times h$
· ReLU activation
· 3D Convolution, c channels, $kernel\_size = t \times 1 \times 1$
· Batch Normalization
· ReLU activation
· Dropout layer $(p = 0.25)$

We tried using 3D convolutions with kernel size $t \times w \times h$, but a model with convolutions $(1 \times w \times h) + (t \times 1 \times 1)$ performed slightly better in our experiments.

We use *mask* data in two ways. Firstly, it is concatenated to *diff* tensor in channels axis. Secondly, after each convolutional block, all elements of the internal representation of *diff*

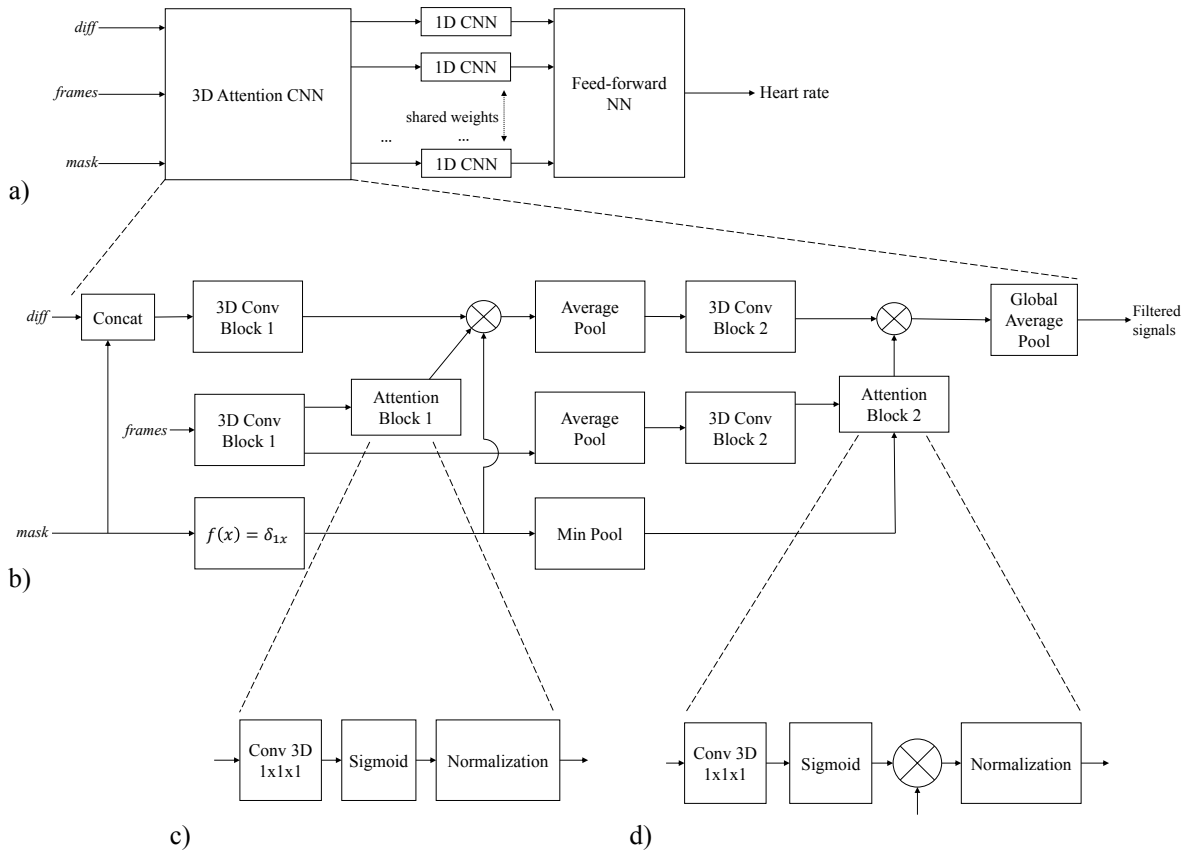


Figure 1: HeartTrack network architecture.

channel are multiplied by zero, if any of the corresponding *mask* values is not equal to 1. This way we are getting rid of possible influence of the background on pulse estimation. In order to choose parts of a face most suitable for pulse tracking at each particular moment, we use attention mechanism (see Figure 1 (c, d)).

To identify the relevant parts of the face, we use original RGB frames of the video since it is commonly acknowledged that they are suitable for detecting face parts and foreign objects. In the end of Attention blocks, we divide attention weights by their mean value over W, H dimensions. This way, the choice of the most relevant fragments does not change the order of values in the network

3.3. Time Series analysis network

In order to obtain the heart rate value from time series extracted from one of the 3D CNN channels, we use 1-dimensional convolutional neural network (1D CNN) with the following sequential architecture:

- Instance Normalization
- 1D Conv, 16 channels, $kernel\ size = 3$
- ReLU activation
- 1D Conv, 16 channels, $kernel\ size = 3$
- Batch Normalization
- ReLU activation
- Max Pooling, $kernel\ size = 2$, $stride = 2$
- 1D Conv, 32 channels, $kernel\ size = 3$, $dilation = 2$
- Batch Normalization
- ReLU activation
- Max Pooling, $kernelsize\ = 2$, $stride = 2$
- 1D Conv, 64 channels, $kernel\ size = 3$, $dilation = 2$
- Batch Normalization
- ReLU activation
- Max Pooling, $kernel\ size = 2$, $stride = 2$
- 1D Conv, 128 channels, $kernel\ size = 3$, $dilation = 2$
- Batch Normalization
- ReLU activation
- Global Max Pooling
- Fully Connected Layer, 30 neurons
- $\tanh$ activation
- Fully Connected Layer, 30 neurons
- $\tanh$ activation
- Fully Connected Layer, 1 neuron

3.4. Synthetic data usage

Speed-up and slow-down video augmentation [22] allows us to synthesize video fragments and corresponding heart rate values for frequencies that are poorly represented in the training data. We also use horizontal flip augmentation during training.

Unlike most computer vision tasks, frequency analysis of temporal signal is critical for the rPPG analysis, while each frame itself does not contain information about the target variable. To address this issue, we have designed a network with a separate 1D CNN part.

We use synthetic data to pre-train the 1D CNN part of the network. Synthetic data does not include video, it is only PPG curves. We sample PPG curves with the following formula:

$$s(t) = A\sin\left(2\pi\int_0^t hr(\tau)d\tau + \phi_{hr}\right) +$$
$$A_2\sin\left(4\pi\int_0^t hr(\tau)d\tau + \phi_{hr}\right) +$$
$$B\sin\left(2\pi\int_0^t br(\tau)d\tau + \phi_{br}\right) + Cn(t),$$

where $hr(\tau)$ is an instantaneous heart rate value, $br(\tau)$ is an instantaneous breath rate value, ϕ_{hr} is an initial phase of the heart cycle, ϕ_{br} is an initial phase of the breath cycle, A is a magnitude of the pulse signal, A_2 is a dicrotic pulse magnitude, B is a breath signal magnitude, $n(\tau)$ is a white noise sample, and C is the standard deviation of the noise.

$hr(\tau)$ can be sampled from a uniform distribution $hr_0 \pm \delta_{hr}hr_0$, where hr_0 is a reference heart rate on the segment, and δ_{hr} refers heart rate variability (we use, $\delta_{hr} = 0.05$). In the same way we introduce breath rate variability parameter $\delta_{br} = 0.1$. Amplitudes of the signals are sampled from uniform distributions $A \sim [0.2, 0.7]$, $A_2 \sim [0, 0.3]$, $B \sim [0.3, 2]$. We use $C = 0.05$.

A sampled curve example is shown at Figure 2.

3.5. Training procedure

First, we perform Xavier initialization [23] with magnitude = 2.34 of all HeartTrack model weights.

After that, we pre-train the 1D CNN network for the task of heart rate value estimation by PPG curve. For this purpose, we synthesise 10^6 PPG curves (i.e. 2222 hours) as described in section 3.4 with reference pulse rate uniformly distributed in [45bpm,180bpm] interval. We use Adam [24] to optimize MSE loss with respect to the 1D CNN model weights. We train the model for 100 epochs with $batch\_size = 32, learning\_rate = 3 \times 10^{-5}$.

And finally, only after all the above procedures we train HeartTrack network end-to-end on video sequences, using Adam optimizer with $learning\_rate$ exponentially decreasing from 10^{-4} to 10^{-5} during 200 epochs with $batch\_size = 16$.

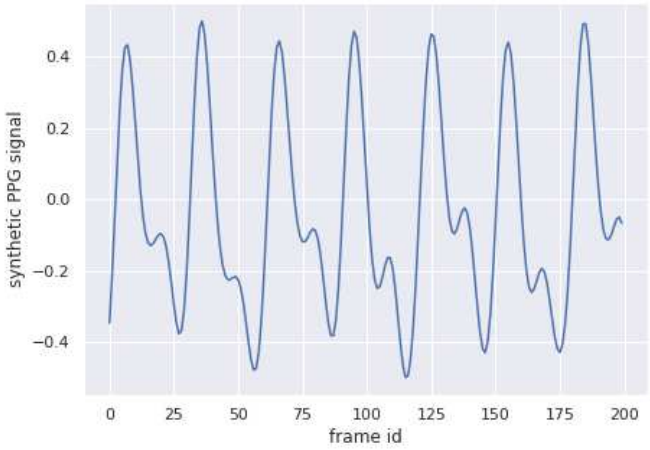


Figure 2: Examples of synthetic photoplethysmography signal with heart rate = 51 bpm.

We have implemented our heart rate estimation pipeline using MXNet framework (https://mxnet.apache.org). HeartTrack network was trained on 1 NVIDIA GeForce GTX 1080Ti GPU. Our HeartTrack implementation can be used for free via API [1].

We believe that our architecture and the way we train the network is specific for the rPPG analysis and heart rate recognition, even though 3D CNN and attention networks are widely used in computer vision.

4. Experiments

4.1. Datasets

We used three datasets for training and testing: two our new databases and the existing database. The first dataset is a **Mo**tion and **Li**ght **p**hoto**p**lethysmo**g**raphy (MoLi-ppg-1) dataset, the second dataset is a MoLi-ppg-2, and the third one is UBFC-RPPG.

Our two new MoLi-ppg-1 and MoLi-ppg-2 rPPG datasets contain complicated and close to natural conditions (movements, speech, different lighting, various equipment, etc). Since at this point the field of rPPG studies is affected by the lack of training data, we believe that these new high-quality datasets themselves will become a valuable contribution to this field.

The first dataset contains 8 hours of video recordings of 30 subjects. The videos were recorded with the following webcams: Logitech C920, Logitech C270, and an HD video camera Canon LEGRIA HFG40. The second

[1] https://api.neurodatalab.dev/

dataset was recorded with different cameras and different subjects. It contains 3,5 hours of video recordings of 15 new subjects. The videos were recorded with a webcam Canyon 720p, and an HD video camera Panasonic. The ground-truth data collected by contact PPG (cPPG) for both datasets was obtained with an optical pulse sensor Shimmer3 GSR+ (www.shimmersensing.com) attached to the subject's finger (sampling rate = 256 Hz), and the data was synced with the video recording. The videos from the webcams were in uncompressed bitmap format with either 800x600 or 1280x720 pixel resolution, and 25 fps. The videos from HD cameras were in uncompressed bitmap format with 1920x1080 pixel resolution and 50 fps. A total of 35 subjects aged 18-35 - both males and females - took part in the experiments. Subjects were lit by fluorescent ceiling lamps and sat in front of the cameras at a distance of about 1 m.

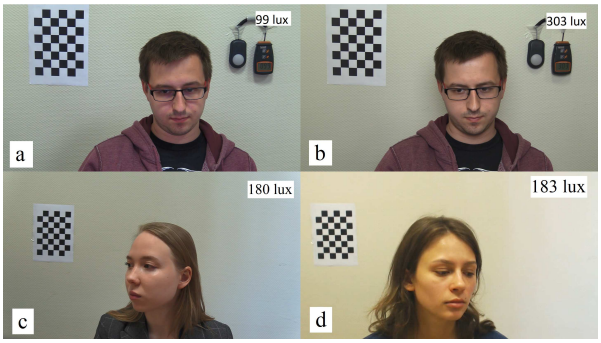


Figure 3: Snapshots of the MoLi-ppg dataset videos. a - frame from video with "cartoon" settings (HD camera), b - frame from video with "light" settings (HD camera), c - frame from video with large head movements (HD camera), d - frame from video with large head movements (Logitech C920). Informed consent for publication was obtained from the subjects..

Three different conditions were used in the first dataset (MoLi-ppg-1):

1. **Static.** The subjects were recorded in varying lighting settings (90-300 lux) while they were sitting naturally in front of the webcam. Among the various illumination conditions there were a) only fluorescent ceiling lamps, b) fluorescent ceiling lamps with an additional spotlight, and c) fluorescent ceiling lamps with a turned on monitor with video.

2. **Movements.** Three cases of head motion in standard conditions included large and small movements as well as speech. In the first two cases the subjects were instructed to perform various types of head movements: left-right, up-down and round. The amplitude of these movements (measured from the straight head position) had to be no more than 45 degrees in the task with small head movements and 80 degrees in the one with large head movements. As for the speech subcategory, the participants were asked to sit facing the cameras and read a text out loud without any head movements.

3. **Recovery after physical stress.** To obtain more broad distribution of pulse, each subject was asked to perform 20-30 squats and was recorded immediately after that.

The second dataset (MoLi-ppg-2) also included three categories:

1. **Static.** The subjects were recorded in varying lighting settings (20-300 lux): a) daylight without lamps, b) fluorescent ceiling lamps with an additional spotlight, c) fluorescent ceiling lamps with a turned on monitor with video.

2. **Speech.** This category includes small natural head motion during speech.

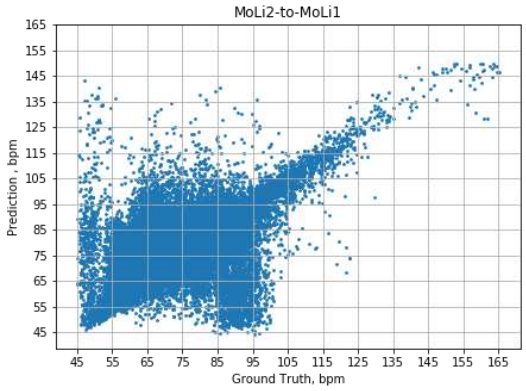
3. **Recovery after physical stress.** Each subject was asked to perform 20-30 squats and was recorded immediately after that, just like in the first dataset.

The public dataset UBFC-RPPG [18] is used to verify the performance of our HeartTrack network. The UBFC-RPPG is specifically designed for the remote pulse rate measurement task. It contains 42 videos from 42 different subjects. The videos were recorded by a Logitech C920HD Pro camera with a resolution of 640x480 in an uncompressed 8-bit RGB format.The participant was asked to play a time-sensitive mathematical game to keep their heart rate varied. The video records natural movements of subjects, including different motions.

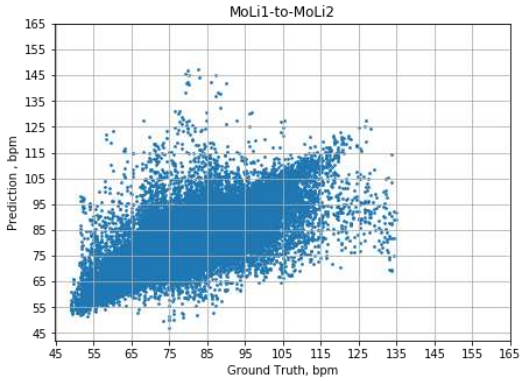
4.2. Evaluation Metrics

To evaluate the performance of our CNN HeartTrack on three databases we used the following metrics:

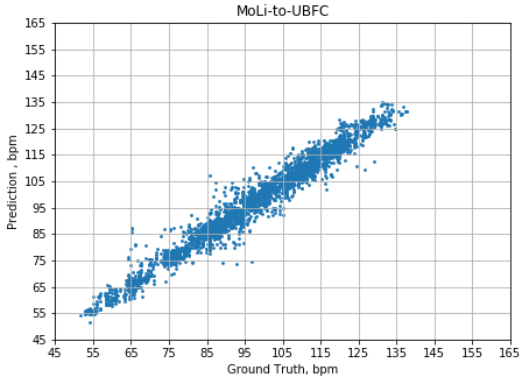
- **Mean Absolute Error (MAE)** in beats per minute (bpm) is calculated as the mean between the pulse obtained from rPPG signals and the pulse obtained from cPPG signals with $\frac{\sum_{v\in\text{videos}}\sum_{k=1}^{T_v}|rPPG_{v,k}-cPPG_{v,k}|}{\sum_{v\in\text{videos}}T_v}$, where T_v is the number of frames in the video v.
- **Root mean square error (RMSE)** = $\sqrt{\frac{\sum_{v\in\text{videos}}\sum_{k=1}^{T_v}(rPPG_{v,k}-cPPG_{v,k})^2}{\sum_{v\in\text{videos}}T_v}}$.
- **Pearson correlation coefficient (r)** = $\frac{\sum\left[(rPPG_{v,k}-\frac{\sum rPPG_{v,k}}{T_v})(cPPG_{v,k}-\frac{\sum cPPG_{v,k}}{T_v})\right]}{\sqrt{\sum\left(rPPG_{v,k}-\frac{\sum rPPG_{v,k}}{T_v}\right)^2\sum\left(cPPG_{v,k}-\frac{\sum cPPG_{v,k}}{T_v}\right)^2}}$, where the sums are taken over all videos v and all the frame ids $k \leq T_v$, in the same way as for MAE and RMSE.



(a) HeartTrack CNN trained on MoLi-ppg-2 and predicted on MoLi-ppg-1 database.



(b) HeartTrack CNN trained on MoLi-ppg-1 and predicted on MoLi-ppg-2 database.



(c) HeartTrack CNN trained on MoLi-ppg-1 and MoLi-ppg-2 and predicted on UBFC-RPPG database.

Figure 4: A scatter plot of ground truth and predicted values.

5. Results and Discussion

We trained our CNN HeartTrack on MoLi-ppg-1 and tested on MoLi-ppg-2, and vise versa. CNN trained on MoLi-ppg-1 showed better quality on the MoLi-ppg-2 dataset (MAE 4.9 bpm, RMSE 7.9 bpm, r 0.8), than trained on MoLi-ppg-2 and tested on MoLi-ppg-1 (MAE 6.4 bpm, RMSE 10.6 bpm, r 0.6), Table 1, Figure 4 a, b. Most likely, this is due to the fact that the first database is much larger than the second. In general, both MoLi-ppg databases are challenging since contain various and difficult conditions. MoLi-ppg-1 and MoLi-ppg-2 datasets were collected in September 2019, so these results are baselines for these databases.

Table 1: Result metrics of HeartTrack network on MoLi-ppg-1 and MoLi-ppg-2 datasets (Mean Absolute Error (MAE), Root mean square error (RMSE)).

Train set	Test set	MAE, bpm	RMSE, bpm
MoLi-ppg-1	MoLi-ppg-2	4.901	7.864
MoLi-ppg-2	MoLi-ppg-1	6.446	10.648

Then we trained CNN HeartTrack on the combination of the MoLi-ppg-1 and MoLi-ppg-2, and tested it on the UBFC-RPPG dataset. We compare our results with existing state-of-the-art methods on this database, that we extracted from the literature. Our approach shows better results than existing ones in most metrics (MAE 2.4 bpm, RMSE 3.4 bpm, r 0.98), Table 2, Figure 4 c.

Table 2: Result metrics of different rPPG methods on the UBFC-RPPG dataset (Mean Absolute Error (MAE), Root mean square error (RMSE) and Pearson's correlation coefficient (r)).

Method	MAE, bpm	RMSE, bpm	r
ICA [25]	3.507	8.635	0.908
CHROM [25]	3.435	4.614	0.968
POS [25]	2.436	6.608	0.936
CK [25]	**2.292**	3.803	0.981
3D CNN [16]	5.450	8.640	
HeartTrack (Ours)	2.412	**3.368**	**0.983**

5.1. HeartTrack internal representations exploration

Even though deep learning models are usually considered to be "black boxes", sometimes exploring internal representations can help to understand, how these models work. We built HeartTrack network under the assumption that its 3D CNN part will learn to clean out the information not related to heart rate from RGB channels. However, we have not optimized 3D CNN weights for this denoising task directly. On Figure 5 we visualize the internal representation (embedding) after 3d CNN filtering. This representation has shape $T \times C$ and we have visualized it as C time series of length T each. One can notice that some of these time series reflect periodical nature of the physiological signal, as we expected. Thus, they are much more beneficial for final heart rate estimation task in comparison with the raw RGB signal (see

Figure 6). Therefore, our 3D CNN architecture is well suited for filtering photoplethysmography signals. However, some channels seem to be useless for heart rate estimation and will likely be ignored in consequent fully-connected layers.

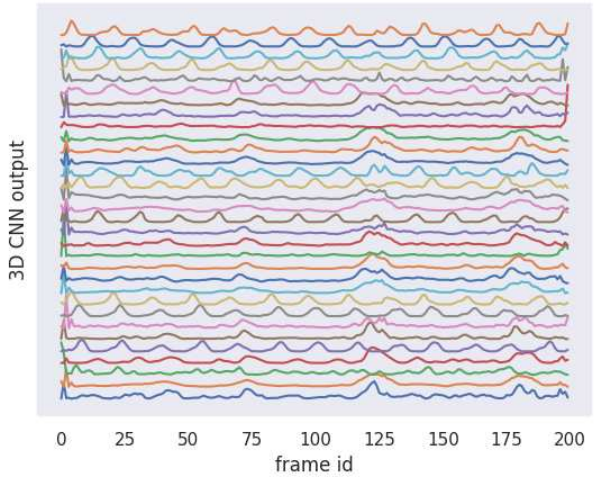


Figure 5: Internal representation of a UBFC video fragment after 3D CNN filtering. Each of the 32 channels is shown as a plot. Constant values were added to each plot for illustrative purposes.

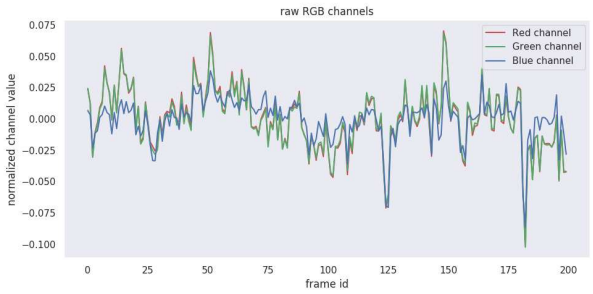
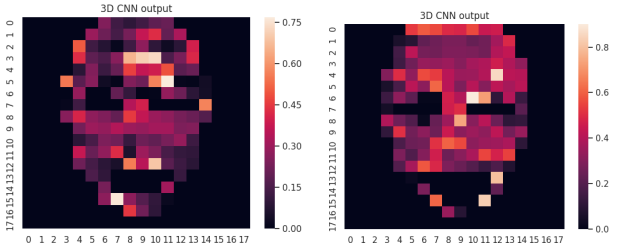


Figure 6: Raw RGB color channels, averaged over facial mask area for a UBFC video fragment.

The effectiveness of 3D CNN as a denoising approach can be explained by the attention mechanism. It allows the model to choose most relevant facial areas using spatial and temporal information. These areas will probably have high signal to noise ratio. These areas will be individual for each person. We visualize attention weights for two subjects from UBFC dataset (Figure 7). As we can see, different areas of the face may be important for different subjects.

However, our method has some limitations. One of them is the distance between the camera and the person as we verified our method only at a distance of 1-1.5 meters. Another limitation of HeartTrack CNN is the need for adequate



(a) Subject 1. (b) Subject 2.

Figure 7: Visualisation of the attention mechanism on one frame for each of the two subjects.

lighting. The lighting level plays an important role in correct pulse detection [26, 27] which suggests that this condition is challenging for remote PPG methods. We intend to investigate this issue in more detail, in particular to test our model in more complex conditions such as different distance between the camera and the subject, various color temperature, quantity, quality, and position of light sources. Furthermore, in addition to external conditions, in the future it is necessary to study the quality of HeartTrack CNN's work on people of different races and ages.

6. Conclusion

In this paper, we describe the HeartTrack neural network for remote heart rate monitoring. The new method was designed to combine the advantages of data synthesis for training with convolutional neural network with attention mechanisms. Our method was tested on three datasets: the public UBFC-RPPG dataset and two introduced datasets, MoLi-ppg-1 and MoLi-ppg-2. Our HeartTrack neural network has shown state-of-the-art results on the UBFC-RPPG database (MAE=2.412 bpm, RMSE=3.368 bpm, R=0.983). The analysis of the results obtained has confirmed that the approach that includes CNN and data synthesis is a promising method for heart rate tracking using real-life data. Furthermore, here we provide a baseline solution for our new datasets that was obtained using the described HeartTrack network. We provide open access to these databases for the research community. In the future, we plan to improve our method using super-resolution neural network and by increasing videos' fps with intermediate frame synthesis as preprocessing steps for heart rate variability estimation.

References

[1] G. Casalino, G. Castellano, V. Pasquadibisceglie, and G. Zaza, "Contact-less real-time monitoring of cardiovascular risk using video imaging and fuzzy inference rules," *Information*, vol. 10, no. 1, p. 9, 2019. 1

[2] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiological measurement*, vol. 28, no. 3, p. R1, 2007. 1

[3] X. Chen, J. Cheng, R. Song, Y. Liu, R. Ward, and Z. J. Wang, "Video-based heart rate measurement: Recent advances and future prospects," *IEEE Transactions on Instrumentation and Measurement*, 2018. 1, 2

[4] M. Kumar, A. Veeraraghavan, and A. Sabharwal, "Distanceppg: Robust non-contact vital signs monitoring using a camera," *Biomedical optics express*, vol. 6, no. 5, pp. 1565–1588, 2015. 1, 2

[5] S. Tulyakov, X. Alameda-Pineda, E. Ricci, L. Yin, J. F. Cohn, and N. Sebe, "Self-adaptive matrix completion for heart rate estimation from face videos under realistic conditions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2396–2404, 2016. 1, 2

[6] X. Liu, X. Yang, J. Jin, and J. Li, "Self-adaptive signal separation for non-contact heart rate estimation from facial video in realistic environments," *Physiological Measurement*, vol. 39, 06 2018. 1, 2

[7] Z. Yang, X. Yang, J. Jin, and X. Wu, "Motion-resistant heart rate measurement from face videos using patch-based fusion," *Signal, Image and Video Processing*, pp. 1–8, 2019. 1, 2

[8] 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)*, pp. 3580–3585, IEEE, 2018. 1, 2

[9] R. Spetlik, J. Cech, V. Franc, and J. Matas, "Visual heart rate estimation with convolutional neural network," 08 2018. 1, 2

[10] W. Chen and D. McDuff, "Deepphys: Video-based physiological measurement using convolutional attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 349–365, 2018. 1, 2

[11] G. De Haan and V. Jeanne, "Robust pulse rate from chrominance-based rppg," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2878–2886, 2013. 1

[12] 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*, vol. 64, no. 5, pp. 1003–1010, 2016. 1

[13] N. A. Fenske and C. W. Lober, "Structural and functional changes of normal aging skin," *Journal of the American Academy of Dermatology*, vol. 15, no. 4, pp. 571–585, 1986. 1

[14] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000. 2

[15] M. Lewandowska, J. Rumiński, T. Kocejko, and J. Nowak, "Measuring pulse rate with a webcam—a non-contact method for evaluating cardiac activity," in *2011 federated conference on computer science and information systems (FedCSIS)*, pp. 405–410, IEEE, 2011. 2

[16] F. Bousefsaf, A. Pruski, and C. Maaoui, "3d convolutional neural networks for remote pulse rate measurement and mapping from facial video," *Applied Sciences*, vol. 9, no. 20, p. 4364, 2019. 2, 7

[17] H. Lee, S. Eum, and H. Kwon, "Is pretraining necessary for hyperspectral image classification?," *arXiv preprint arXiv:1901.08658*, 2019. 2

[18] S. Bobbia, R. Macwan, Y. Benezeth, A. Mansouri, and J. Dubois, "Unsupervised skin tissue segmentation for remote photoplethysmography," *Pattern Recognition Letters*, vol. 124, pp. 82–90, 2017. 2, 3, 6

[19] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollàr, "Focal loss for dense object detection," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 2999–3007, 2017. 3

[20] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017. 3

[21] X. Dong, S.-I. Yu, X. Weng, S.-E. Wei, Y. Yang, and Y. Sheikh, "Supervision-by-Registration: An unsupervised approach to improve the precision of facial landmark detectors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 360–368, 2018. 3

[22] X. Niu, H. Han, A. Das, A. Dantcheva, X. Chen, and X. Z. Shiguang Shan, "Robust remote heart rate estimation from face utilizing spatial-temporal attention," *IEEE AFGR 2019 Conference paper*, 2019. 5

[23] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010. 5

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 5

[25] R. Song, S. Zhang, J. Cheng, C. Li, and X. Chen, "New insights on super-high resolution for video-based heart rate estimation with a semi-blind source separation method," *Computers in Biology and Medicine*, 11 2019. 7

[26] R. Amelard, C. Scharfenberger, F. Kazemzadeh, K. J. Pfisterer, B. S. Lin, D. A. Clausi, and A. Wong, "Feasibility of long-distance heart rate monitoring using transmittance photoplethysmographic imaging (ppgi)," *Scientific reports*, vol. 5, p. 14637, 2015. 8

[27] J. Przybyło, E. Kańtoch, M. Jabłoński, and P. Augustyniak, "Distant measurement of plethysmographic signal in various lighting conditions using configurable frame-rate camera," *Metrology and Measurement Systems*, vol. 23, no. 4, pp. 579–592, 2016. 8