IM-TWIN: from Intrinsic Motivations to Transitional Wearable INtelligent companions for autism spectrum disorder

a European funded project

Processing of physiological signals, visual info, and PlusMe interaction, second version

Deliverable 2.2

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952095.

Project duration 36 months (November 2020, October 2023). Consortium: Consiglio Nazionale dele Ricerche (ITA), Universiteit Utrecht (NLD), Centre de Recherches Interdisciplinaires (FRA), Università degli Studi di Roma La Sapienza (ITA), Plux-Wireless Biosignals S.A. (PRT).

Deliverable data

Work Package:	2 Affective signal processing through the integration of multiple sources
Work Package leader:	UU
Deliverable beneficiary:	UU
Dissemination level:	public
Due date:	31th July 2023 (Month 33)
Type:	Report
Revision:	2 (March 2024)
Authors:	Lukas P.A. Arts, E.L. van den Broek, Francesco Montedori, Valerio Sperati, Massimiliano Schembri, Gianluca Baldassarre

Acronyms of partners

CNR-ISTC	Consiglio Nazionale delle Ricerche, Istituto di Scienze e Tecnologie della Cognizione (Italy)
UU	Universiteit Utrecht (The Netherlands)
CRI	Centre de Recherches Interdisciplinaires (France)
LA SAPIENZA	Università degli Studi di Roma La Sapienza (Italy)
PLUX	Plux - Wireless Biosignals S.A. (Portugal)

Table of contents

1. Overview of the deliverable

This deliverable contains the final version of the processing and feature extraction pipeline for the biosignals and visual information. The core of the deliverable is sectioned as follows:

2. *Processing of physiological signals,* containing the subsections:
   - 2.1. *Signal Quality Indicator (SQI)*, describing the improvements made to the previous SQI in order to lay a solid foundation for further processing.
   - 2.2. *ECG*, describing the segmentation, peak extraction, and high-level feature extraction of the ECG signal.
   - 2.3. *EDA*, describing a wavelet-based denoising filter and presenting a novel wavelet-based feature extraction technique to handle the severe noise bursts.
3. *Processing of visual information,* describing the *Graphic User Interface* (GUI) of the "eye contact detector" software, developed to facilitate the use of the tool by the researchers and the subsequent data analysis.
4. *Processing of interaction between child, PlusMe and therapist*, describing the software improvement which allows the automatic synchronization of the two logs recorded by the TWC toy (data about toy manipulation) and the *camera glasses* (data about eye contact detection between child and therapist).

2. Processing of physiological signals

Processing the physiological signals from the IM-TWIN T-Shirt is inherently complex. As detailed in previous reports, these signals are prone to distortion, and the frequent shifting of electrodes can result in signal loss. Consequently, biosignal processing within the IM-TWIN ecosystem is far from straightforward.

As the quality of the processing's output is directly tied to the quality of the input data, a signal quality indicator (SQI) was developed (refer to report D2.1). This indicator employs a binary decision filter that screens out signals unsuitable for processing. Its role is crucial in the processing pipeline, enhancing reliability and enabling the system to respond when it's unable to infer an affective state. This is particularly vital in therapeutic or medical settings where using AI requires caution [1]. In these contexts, a system that says it doesn't know is preferred over a system that hallucinates incorrect conclusions [2].

Mainly due to the SQI's essential function and feedback from CRI and CNR, we embarked on an in-depth examination of its performance. By manually labeling signals from 8 recordings, we created a ground truth dataset for comparing the output of the SQI. Based on these insights, we formulated an improved version of the indicator: SQI v2.

In this section, we will first discuss the previous SQI's performance and introduce its refined version 2. Next, subsection 2.2 will outline the process of extracting various HR and HRV features from the ElectroCardioGram (ECG) signal, which are known to correlate with affective states. These features will form the basis for affective state classification, as described in report D3.2. The section concludes with a section that delves into the quality and usability of the ElectroDermal Activity (EDA) signal, the techniques developed to denoise the signal and the methods used to extract meaningful features.

2.1 Signal Quality Indicator (SQI)

Traditionally, biosignals, whether obtained from clinically validated sensors or wearable devices, are first processed by a preprocessing and feature extraction pipeline [3]. This is followed by a classification pipeline, which transforms the extracted features into a final classification. Within the context of the IM-TWIN system, this classification would correspond to one of three specific affective states: low arousal, high positive arousal, and high negative arousal.

Some may contend that modern pipelines integrate both processes by employing deep learning architectures capable of self-directed preprocessing and feature extraction [4]. Despite this innovation, the essential function of this processing system remains unchanged: it takes biosignals as input and produces a label belonging to a predetermined set of labels. In settings where high-quality signals with minimal noise are assured, this system functions reasonably well. The IM-TWIN system, however, does not conform to these ideal conditions. Consequently, employing such a processing pipeline in the IM-TWIN context would lead to many false positives and inconsistent behavior. This unreliability stems from the system's susceptibility to various forms of noise and signal loss, hindering its effectiveness in an environment where precision required.

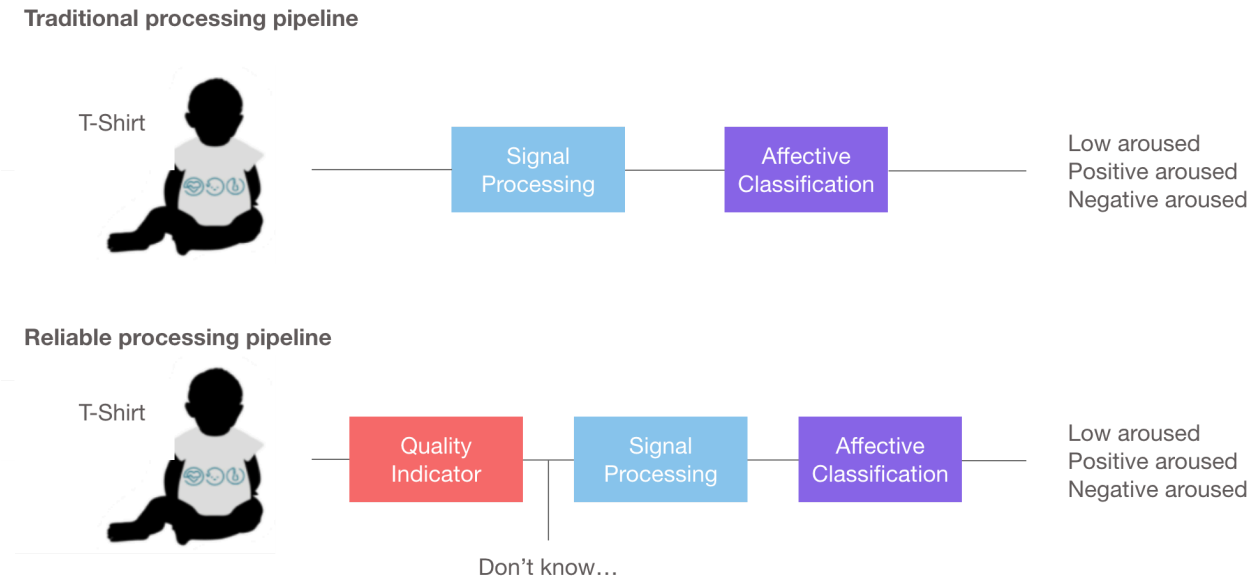

Figure 1: Adding a Signal Quality Indicator to the processing pipeline not only increases accuracy, it also gives the system the ability to express 'I don't know' whenever it cannot infer affective state due to data loss or bad signal quality.

To address the challenges faced by the IM-TWIN system, we created a Signal Quality Indicator (SQI). The primary purpose of the SQI was to provide therapists with clear, actionable feedback on signal quality, enabling them to adjust the fitting of the T-Shirt as needed (refer to report D2.1). Beyond this immediate application, the SQI introduced a significant addition to the classification pipeline: the addition of a fourth output, essentially signaling "I don't know." representing that the source data might be unprocessable due to its low quality (see Figure 1). The ability of a system to articulate its uncertainty has been shown to enhance users' trust [5]. As a result, the SQI has become central to the reliability and trustworthiness of the entire processing pipeline. However, its sequential nature also means that any error made by the SQI inevitably translates into a subsequent mistake. Thus, the performance of the SQI is of utmost importance.

Regrettably, the mask-based SQI as detailed in report D2.1 did not meet our expectations in terms of accuracy. Moreover, further feedback from CRI and CNR prompted us to undertake a more comprehensive examination of the SQI's performance.

It should be noted that the SQI only takes the ECG signal as input. This is a deliberate decision as ECG is a more sensitive and faster responding biosignal than EDA. If good quality ECG can be obtained, we can safely assume this to be the case for EDA as well. This assumption can be further supported by the fact that the majority of the noise experienced in IM-TWIN comes from large body movements and that the electrodes of both signals are placed very close to each other. As such, movement artifacts in one signal are highly correlated to artifacts in the other biosignal.

2.1.1 SQI v1.0

To test the performance of the mask-based SQI we designed a quantitative, repeatable benchmark experiment. First, a labeled dataset was generated by manually inspecting 8 recordings recorded by CRI in April 2023. To speed up labeling, a specially designed labeling tool was used, called LabelStudio [6]. LabelStudio provides a user-friendly interface for the labeling of various data types including time series (see Figure 2).

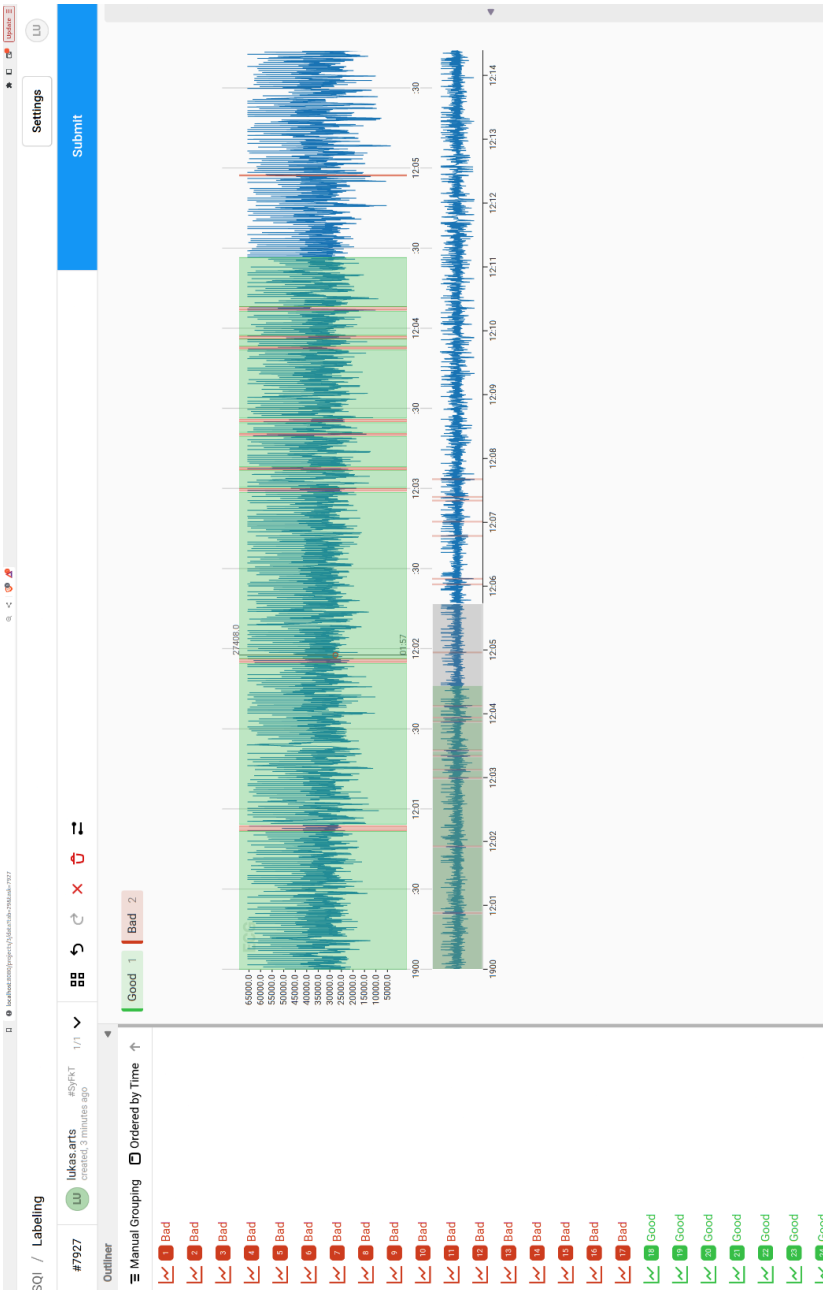

Figure 2: Label Studio was used to speed up manual quality labeling of the ECG records.

Initially, we conducted a rough labeling pass using LabelStudio, focusing specifically on the usability of the signal. We categorized consecutive regions of the signal as either 'good' or 'bad'. Any instances of signal loss were automatically labeled as 'bad', and if a signal was too distorted to extract QRS peaks, it was also deemed 'bad'. Conversely, when the signal was noisy but the QRS peaks were still distinguishable from the background noise, we labeled it 'good'. Our aim was not to attain a 'perfect' ECG signal. Instead, we concentrated on the feasibility of extracting QRS peaks for HR and HRV analysis.

We then divided the labeled signal into 3650 two-second chunks, marking any chunk as ‘bad’ that was partly or entirely within a 'bad' region, and labeling the remaining as 'good'. The labeling process then advanced to the first inspection round, where we used a custom tool that enabled rapid and efficient examination of all 3650 labeled chunks. We displayed chunks and their labels ten at a time, across 365 pages. The tool then allowed easy label modification through visual interaction with the chunk. We repeated this correction procedure three times, shuffling the order and taking at least one-hour breaks for refreshment. After three days, we conducted a final inspection of the chunks to ensure maximum quality.

Subsequently, we applied the mask-based SQI to the same eight recordings, dividing its binary output into 2-second chunks as well. We then compared the true and predicted labels for each chunk, calculating the precision, recall, and F1-score (see Table 1). Precision assesses how accurately the samples labeled as good or bad reflect their actual status, while recall measures the number of correct samples retrieved by the classifier. The F1-score represents the harmonic mean between these two measures. Additionally, we computed the confusion matrix (see Figure 3), providing further insight into the performance of our labeling approach.

Table 1: Performance metrics of the mask-based SQI (SQI v1) on a manually labeled testset composed of 8 recordings.

Labels	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**Support**
Good	86.5	64.1	73.6	1350 chunks
Bad	81.7	94.1	87.5	2300 chunks
Total	**84.1**	**79.1**	**80.0**	

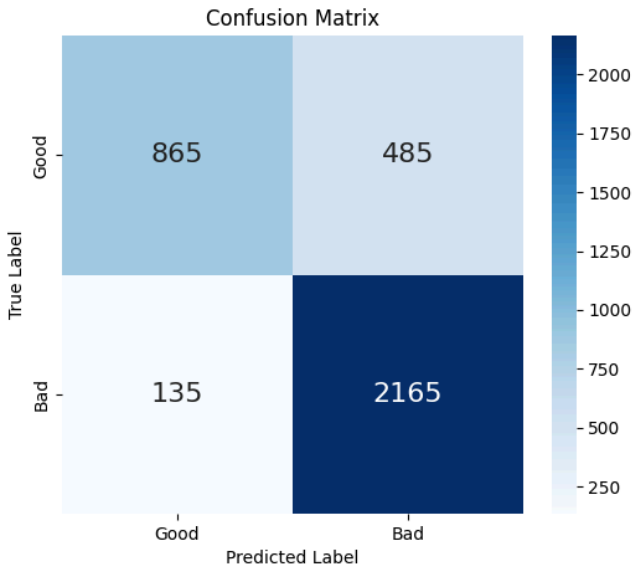

Figure 3: Confusion matrix of the mask-based SQI (SQI v1) on a manually labeled testset composed of 8 recordings.

The mask-based SQI achieves an F1-score of 80% on the manually labeled test set. This result emerges from a combination of a 74% F1-score for the 'good' samples and an 88% F1-score for the 'bad' quality samples. In essence, this outcome indicates that the classifier tends to be somewhat 'too strict,' often categorizing 'good' samples as 'bad'. More precisely, the SQI fails to retrieve 35% of the 'good' samples, meaning that a significant portion of data suitable for further processing is discarded. Furthermore, of the remaining data, 15% is deemed unusable. This segment contributes to false positives, thereby diminishing the overall reliability of the system.

In sum, while the mask-based SQI shows substantial effectiveness in identifying 'bad' quality samples, its stringent criteria lead to an over-rejection of 'good' samples. This highlights a critical area for potential improvement. Enhancing the SQI's accuracy could substantially improve the accuracy and trustworthiness of the entire system.

2.1.2 SQI v2.0

The necessity for a more precise Signal Quality Indicator (SQI) is clear. However, designing a high-quality SQI is a more complex task than it might appear at first glance. While human experts can readily determine signal usability after minimal training, the same task becomes exceedingly difficult for a computer, particularly when the signal is distorted to the point of it being barely usable.

Our previous SQI often chose the side of caution, classifying samples around edge cases as 'not usable' to play it safe. Yet, we do not seek perfect ECGs; we merely need to detect peaks to

extract HR and HRV, meaning that noise is acceptable to a certain degree. Determining this exact level of acceptable noise is a hard for three main reasons:

1. **Noise Source Difference:** The sources of noise encountered with the IM-TWIN T-shirt differ greatly from those found with wet, clinically validated electrodes. While clinical ECGs often struggle with baseline wander and powerline interference, the IM-TWIN system faces severe, short-burst distortions due to electrodes losing contact with the skin.
2. **Design-Correlated Noise:** The specific noise we encounter is strongly tied to the design of the IM-TWIN T-shirt, meaning standardized, pre-built SQIs are not compatible with our data. In our previous version, we tried to solve this by targeting different noise types, but issues like noises caused by electrode shift, which occupy the same frequency band as the desired QRS peaks, proved challenging to detect.
3. **Noise Tolerance:** Our system permits more noise than most other ECG processing systems, making off-the-shelf SQIs unsuitable due to their overly strict criteria.

To enhance the previous mask-based SQI, we adopted a radically different approach utilizing a proven technique often applied to ECG classification tasks: the deep learning residual network (ResNet) architecture [7]. Instead of relying on predefined heuristics, we trained the model by showing it many examples of 'good' and 'bad' quality signal fragments. This method, first introduced by Hannun et al. in Nature Medicine [8], has gained considerable attention for various ECG classification tasks.

While Hannun et al.'s approach classified 8 different types of cardiac arrhythmia, we adapted it for our purposes as a binary quality classifier, replacing the multi-class classification head with a binary one. An overview of this innovative deep learning architecture is depicted in Figure 4.

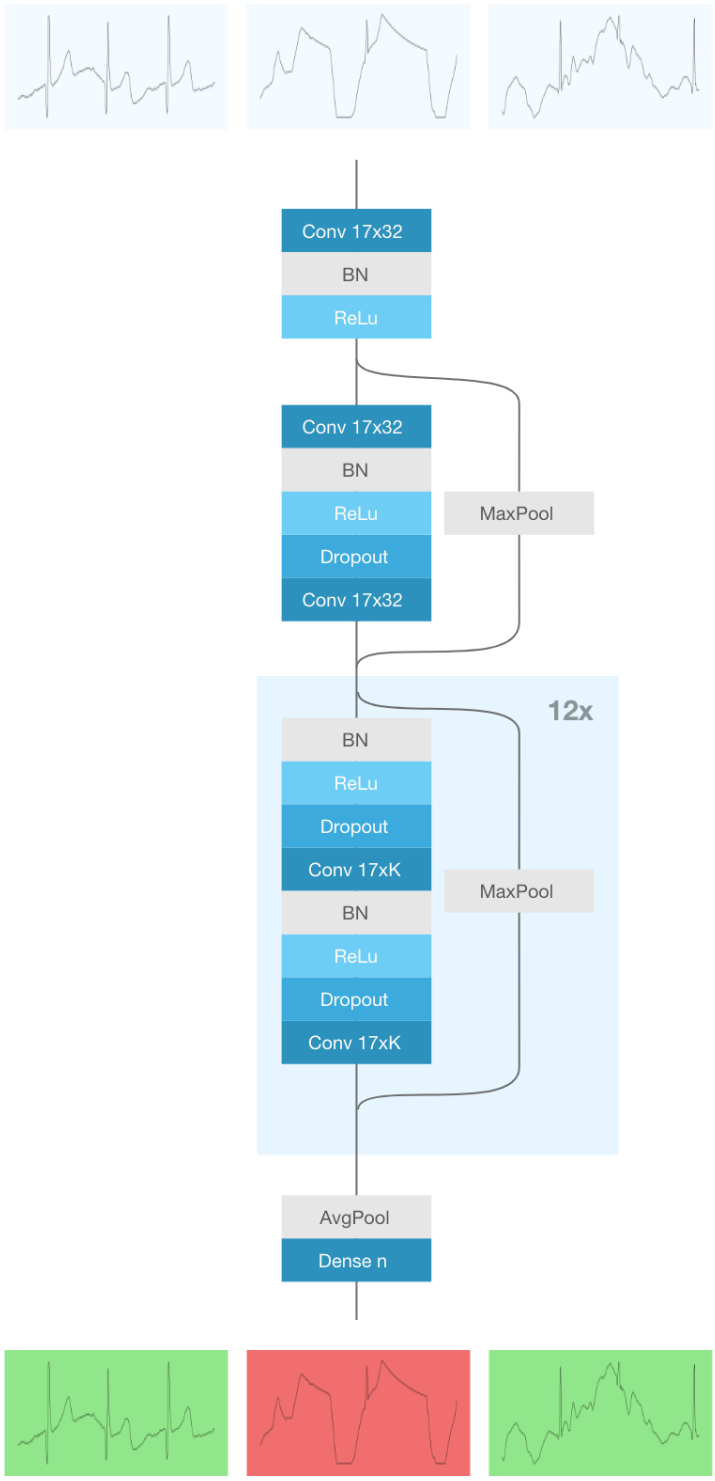

Figure 4: Adapted ResNet architecture as proposed by Hannun et al [8] for assessing signal quality of 2-second ECG fragments.

Hannun's ResNet architecture stands out for its remarkable improvement over traditional CNNs, particularly when it comes to handling the learnability problem. Traditional CNNs tend to struggle with learning when they reach 10 or 20 layers. ResNet, however, addresses this by incorporating skip connections (represented as MaxPool blocks in Figure 4) [7]. These connections facilitate data flow even when hundreds of layers are involved, enabling ResNet to detect smaller, more subtle details in the data. This leads to a significant performance boost for tasks requiring nuanced understanding. Although other deep learning techniques for time series classification have been developed (e.g., transformers [9], autoencoders [10]), the ResNet architecture is ideal for our specific needs. In a temporal sense, we have very short samples of only 2 seconds. Hence, we don't require classification over extensive temporal distances. However, these 2 second chunks each contain a 1000 data points in which we search for three or four spikes of which we don’t know the actual shape. As such, we need a network capable of learning these intricate differences between time series. For this, the ResNet model is perfect.

The data processing follows the previously described procedure. Segmented into 2-second chunks, the data provides enough information to span 3 or 4 heartbeats while remaining short enough to maintain near-real-time performance. We then divided the chunks, designating 50% for training and the remaining 50% for testing. Training was executed using the Adam optimizer [11] with a learning rate of 0.001 and a "reduce-on-plateau" learning schedule, which reduces the learning rate by a factor of 10 if the validation loss doesn't decrease for two consecutive epochs. Table 2 and Figure 5 show the performance of the improved SQI, displaying the confusion matrix along with the precision, recall, and F1 score for the portion of the dataset set aside for testing.

Table 2: Performance metrics of the ResNet-based SQI (SQI v2) on a manually labeled testset composed of 8 recordings.

Labels	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**Support**
Good	93.5	95.5	94.5	651 chunks
Bad	97.5	96.3	96.9	1174 chunks
Total	**95.5**	**95.9**	**95.7**	

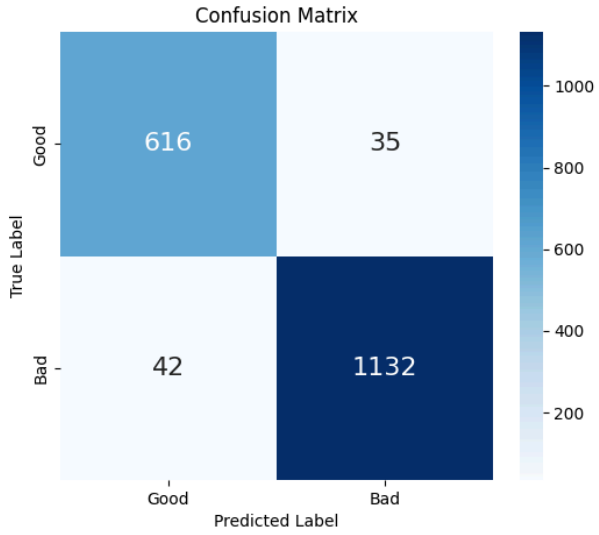

Figure 5: Confusion matrix of the ResNet-based SQI (SQI v2) on a manually labeled testset composed of 8 recordings.

The newly developed SQI exhibits a significant improvement in F1-score of 95.7%, marking an approximately 15% improvement over the previous version. This result proves ResNet's ability to learn the tiny variances found in our data. The most notable improvement is reflected in the recall score for samples labeled as 'good.' Whereas the former SQI overlooked 35% of the 'good' samples, the updated SQI misses only 4.5%—a seven-fold improvement. Furthermore, this enhancement in recall does not come at the expense of precision, meaning that the update genuinely advances upon the prior SQI. Figure 6 illustrates two segments that were challenging for SQI v1 to differentiate due to their nearly identical frequency spectrum. Nevertheless, the upper segment reveals the presence of three QRS-shaped peaks, permitting the extraction of HR and HRV features, while the lower segment displays only one peak.

Good *Bad*

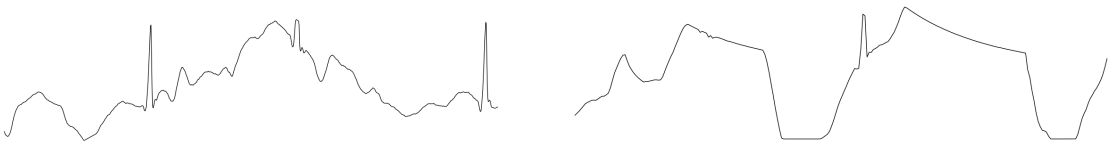

Figure 6: A 'good' quality (left) and 'bad' quality (right) segment showing the difficulty in distinguishing usable from unusable ECG. Both samples have sharp peak-like features that would be hard to distinguish based on a frequency spectrum alone. At the same time, a matching filter would be too strict as the QRS peaks are sometimes severely distorted.

2.2 ECG

Following an inspection of signal quality, the signal processing pipeline splits into two branches; one focuses on the ECG signal, and the other on the ElectroDermal Activity (EDA) signal. In this segment, we describe the ECG signal's segmentation, the extraction of its peaks, and the calculation of several heart rate (HR) and heart rate variability (HRV) features. The EDA processing branch will be described in Section 2.3.

2.2.1 ECG preprocessing

The IM-TWIN system's essential capability is to conduct real-time analyses. Biosignals must be processed fast and affective state classification must occur near instantaneously. Regrettably, numerous ECG features that correlate with the autonomic nervous system's state, and therefore the child's affective state, cannot be calculated instantaneously [12]. Often, they need a distribution of interbeat intervals (IBIs) (e.g., the time between two successive heartbeats) of at least 2-5 minutes of high-quality ECG data [13]. Thus, ECG features more or less provide an averaged depiction of one's biosignals over the past 2-5 minutes. Instantaneous ECG features are unfeasible. Fortunately, Laborde et al. [24] suggested shorter time windows for time-domain HRV metrics. Munoz et al. even claimed that 2-minute windows were unnecessary and that near perfect results could be obtained using 30s windows. However, frequency-based HRV metrics still need at least 1-2 minutes of ECG to make accurate frequency estimations. Therefore, we opted to segment the ECG using two windows; a 30s window for time-based HRV and a 120s window for frequency-based HRV. The 120s window had a 30s stride to match the output of the 30s window (see Figure 7).

Upon segmentation, the window's signal quality is examined. Short periods of noise are tolerable as they have minimal effect on the overall feature extraction. Nevertheless, this tolerance has a limit. Eventually, there is too little data for a trustworthy estimate. Therefore, we establish a threshold of a minimum 80% high-quality ECG data in each window. If this ratio falls below 80%, the system ceases making affective state predictions and defaults to an 'I don't know' status until the quality is reestablished. A threshold of at least 80% high-quality data is chosen to balance the quality and quantity of output. A threshold too high would result in many undesirable ‘I don’t know’ states. However, a threshold set too low endangers the system’s credibility. Figure 8 illustrates the quality assessment in operation.

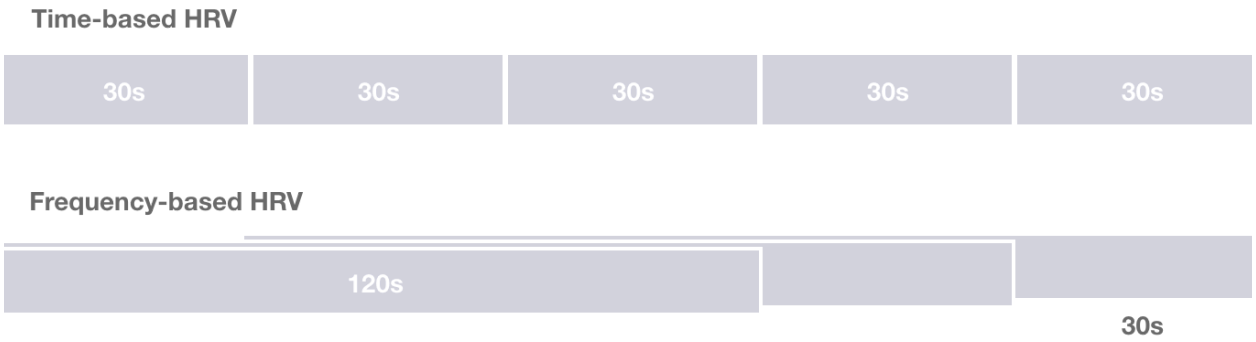

Figure 7: ECG is analyzed using a 30s window and a 120-second trailing window having a stride length of 30 seconds for time and frequency-based HRV metrics, respectively. The window trails analysis instead of surrounding the analysis. The latter would require looking into the future which would impair real-time analysis.

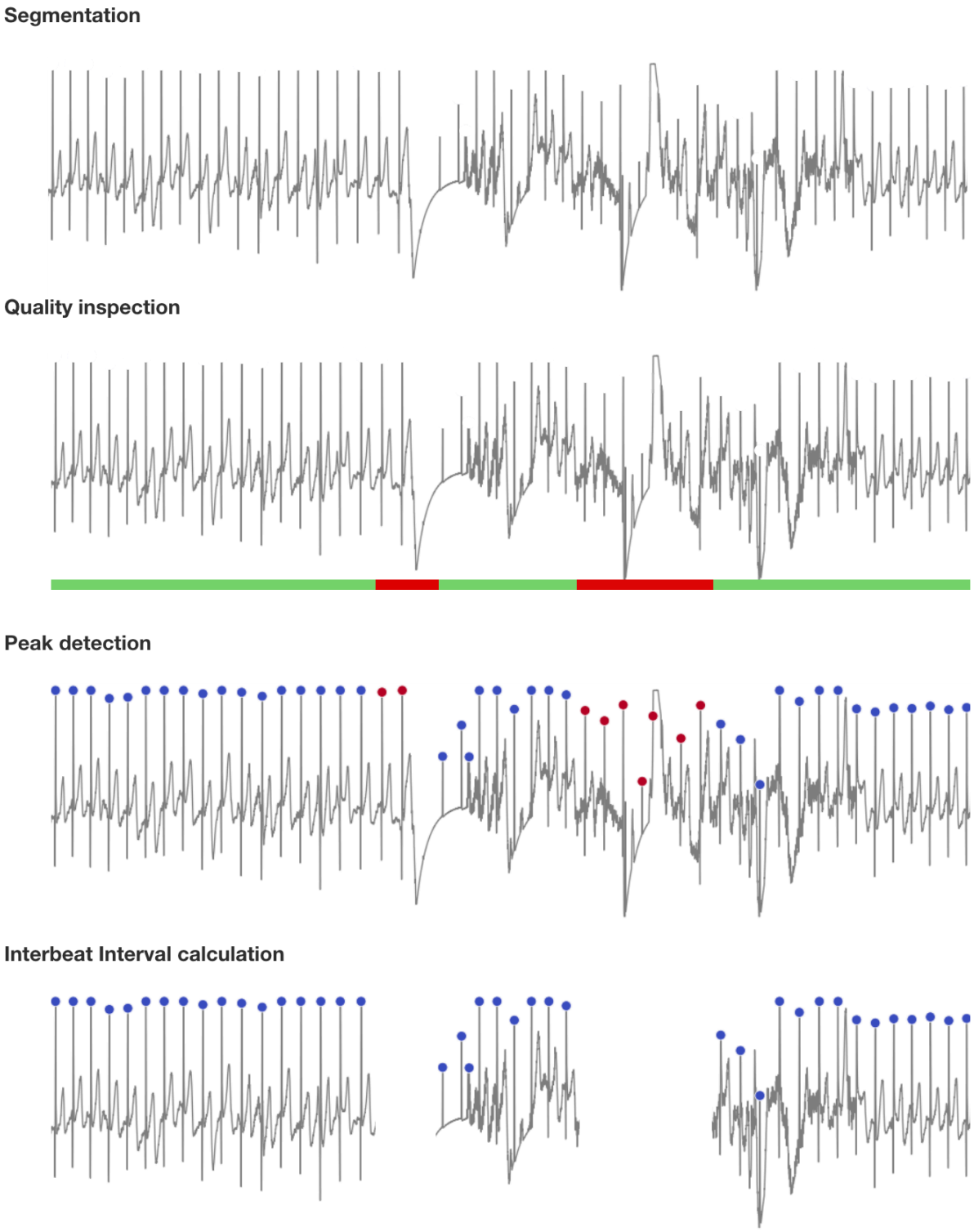

Figure 8: Major steps in the ECG processing pipeline. First, the signal is segmented. Subsequently, quality is assessed by the SQI. If more than 80% of the segment contains good data, peaks are extracted. Finally, peaks are marked unreliable if they reside in bad quality signal regions. Peaks marked reliable are used for further feature extraction.

2.3.2 ECG feature extraction

When a window fulfills the quality criteria, the processing pipeline advances to peak extraction. Over time, many ECG peak detectors have been developed, with a famous one developed by Pan and Tompkins [14] standing as one of the most favored today. Created in 1985, this detector demonstrates remarkable reliability in clinical environments. However, the IM-TWIN signals are noisy. Even after quality evaluation, the ECG signal might still include substantial noise bursts. As detailed in report D3.1, the Pan and Tompkins detector tends to become unreliable when confronted with noisy ECG signals. This is a significant issue, as many HRV features are extremely sensitive to errors in peak detection [15]. A single incorrect peak can influence the feature extraction across the entire 2-minute window.

To mitigate the risk of such errors, we utilized a recently validated machine learning-based method. This approach, developed by Zahid et al. [16], leverages a deep Convolutional Neural Network (CNN) trained on extensive hours of noisy Holter recordings, such as wearable ECGs. By employing Zahid et al.'s peak detector in combination with our precise SQI, we created a highly resilient peak detection pipeline. It's well-equipped to manage the often unexpected and severe bursts of ECG noise that are common in the IM-TWIN system. Figure 8 visually captures both systems at work.

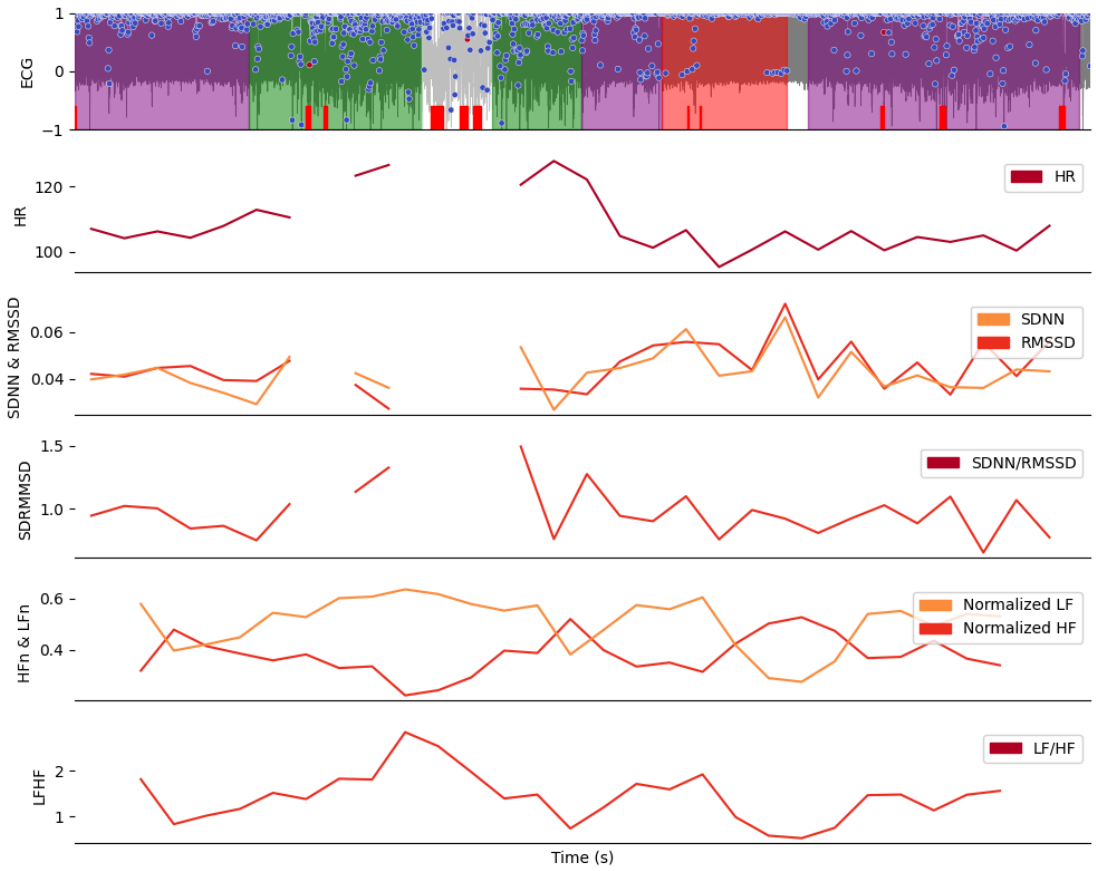

Figure 9: ECG feature extraction for a single recording. Top: Raw ECG signal, the extracted QRS peaks with their quality shown as blue or red points and the child's affective states as annotated by the therapists. Regions of bad quality are also visualized by a red bar. Second, third and fourth from top: Average heart rate based on the extracted QRS peaks. SDNN and RMSSD, two commonly used HRV metrics, calculated from the interbeat intervals of the QRS peaks. Bottom two plots: HRV frequency metrics known to correlate with vagal tone (resting state).

In the final processing step of the ECG processing pipeline, we remove outliers, interpolate and extract HR and HRV features. Suspicious long heart beat intervals are removed using the maximum percentage change rule [25]. An interbeat interval is allowed to differ at most 30% from the mean of the last four intervals. Subsequently, gaps smaller than 3 seconds are interpolated using linear interpolation and the filling technique described by [26]. Lastly, HR and HRV features are extracted. Figure 9 shows average HR and two commonly used metrics for HRV, the standard deviation of subsequent normal-normal peak intervals (SDNN) and the Root Mean Square of Successive Differences (RMSSD) [13]. You should keep in mind that features are calculated based on the last 120 seconds of ECG data. As such, Figure 9 does not show HR or HRV features for the first 120 seconds of data. Figure 9 also shows the considerable impact of noisy ECG measurements around t=800s. A single error shifts both HRV metrics significantly for the duration of an entire window (120s).

As the selection of HR and HRV features is highly task dependent, we discuss this in more detail in report D3.1: Personalized affect classification and feedback.

2.3 EDA

In addition to the ElectroCardioGram (ECG) signal, the IM-TWIN T-Shirt also records ElectroDermal Activity (EDA) with two electrodes placed on the back of the child. EDA measures the skin's conductivity, reflecting the activity of sweat glands [17]. As the sweat gland activity is influenced directly by the sympathetic nervous system [12], EDA is often used as a non-invasive measure of a person's physical arousal level [18]; a quality that makes EDA critically essential to the IM-TWIN system.

The extraction of high-quality features from the EDA signals collected by the IM-TWIN system using traditional techniques was anticipated to be challenging (refer to report D3.1). The report determined that novel strategies needed to be formulated to manage the intense noise bursts observed in the ECG signal, and believed to exist in the EDA signal as well. In this part of the report, we begin by illustrating the EDA signal as captured by the T-Shirt, contrasting it with clinically obtained EDA (e.g., EDA measured from the fingers or palms using wet, adhesive electrodes). We then demonstrate the performance of standard preprocessing methods and explain why they fall short for IM-TWIN. We conclude by presenting the use of an adaptive wavelet-based denoising method and a novel time-frequency technique. These innovations enable us to derive meaningful features from severely distorted EDA signals.

2.3.1 EDA of IM-TWIN

Typically, EDA signals are divided into their tonic and phasic components prior to feature extraction [17]. This division is grounded in the understanding that the EDA signal consists of a superposition of two elements: a slow-varying, large-amplitude conductance level (the skin conductance level (SCL)), and quicker varying, small-amplitude skin conductance responses (SCRs). These two components are distinguishable by frequency, as each reside in a unique frequency band [20]. Figure 10 illustrates this decomposition for a standard EDA signal.

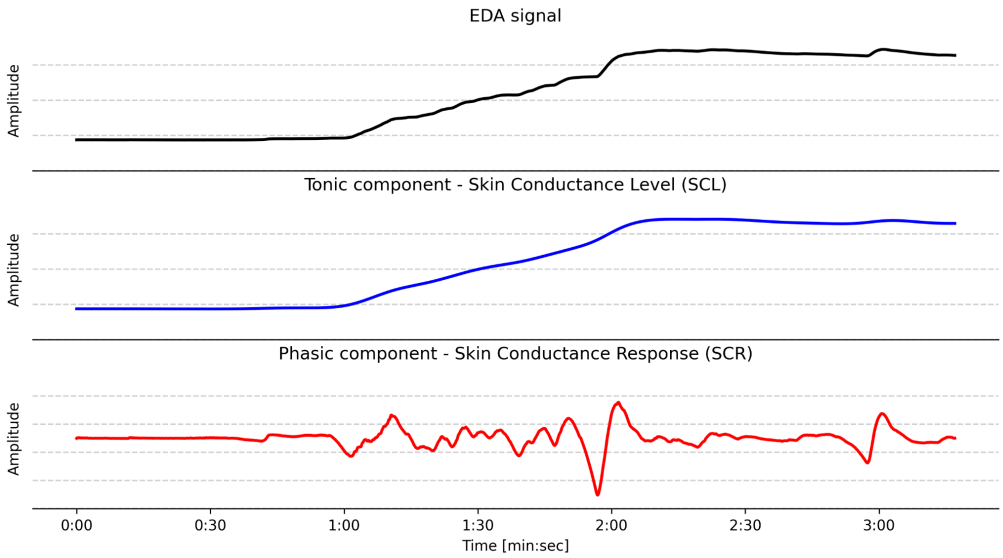

Figure 10: Typical EDA signal and its decomposition into the tonic and phasic components.

The EDA signal obtained by the IM-TWIN T-Shirt is significantly different from this ideal example. Figure 11 shows a 2 minute EDA signal from a high-quality recording. Note, the recording already passed the SQI meaning there was a proper T-Shirt fit and all electrodes had contact with the skin.

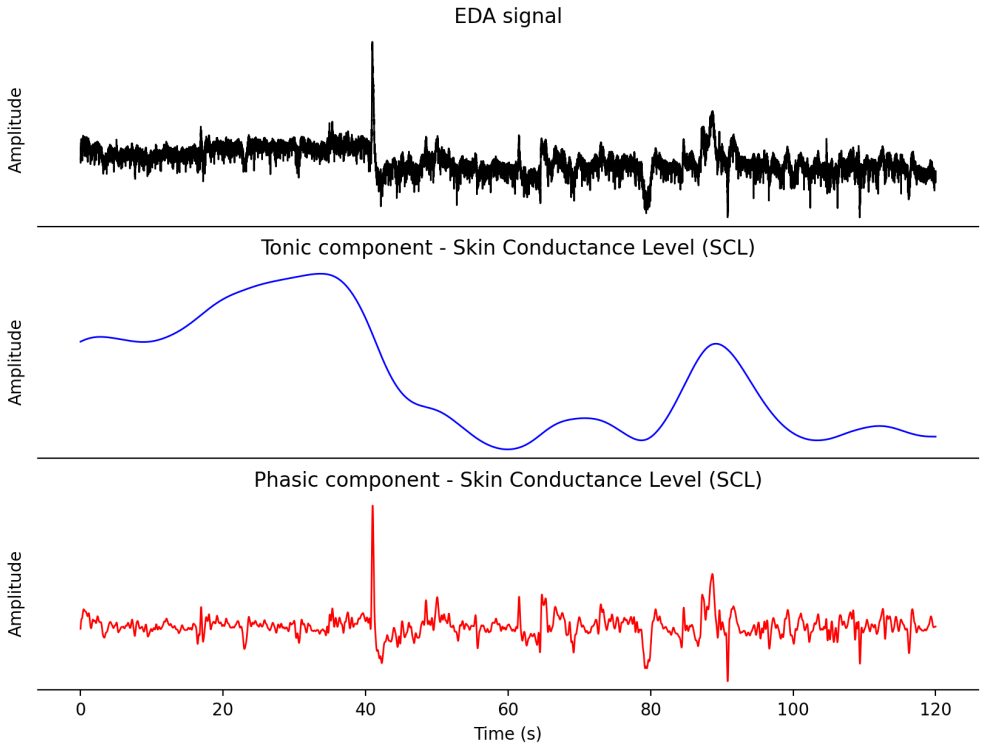

Figure 11: EDA signal as recorded by the electrodes on the back of the IM-TWIN T-Shirt. The selected interval passed the ECG-based quality inspection by the SQI. The differences between

typical EDA and this signal are striking. Where typical EDA is slow moving, the EDA from IM-TWIN shows many rapidly changing, high-frequency components.

The discrepancies between the clinically acquired EDA recording and the one acquired through the IM-TWIN T-Shirt are striking. EDA is generally considered a 'slow' signal, with most of its power situated within the slowly varying [0.05-0.5] Hz frequency range [12]. It's also considered slow in responding to external stimuli. While heart rate may react within seconds, EDA can take up to ten times longer to manifest a response [3]. However, the EDA in IM-TWIN reveals fluctuations at much finer time scales (e.g., higher frequencies). Unlike typical EDA, characterized by quick increases and slow decreases, our EDA does not display this behavior. In some instances, there are even rapid oscillations that are highly uncommon for skin conductance [17]. In reality, such oscillations are more typical in high-frequency recordings of muscle activity (EMG). An explanation for this behavior might lie in the fact that IM-TWIN measures EDA at the child's back, a location that is generally not free from EMG interference. The activity of the back muscles or the tensioning of the back muscles due to breathing could generate the high-frequency EMG interference observed in the recordings.

2.3.2 EDA preprocessing

Because IM-TWIN's EDA signal suffers significantly from noise caused by activity of the muscles (EMG) surrounding the EDA electrodes, the signal needs to be cleaned before further processing can happen. Unfortunately, removing EMG noise from EDA is not as trivial as just applying a low-pass Butterworth filter as is often done in EDA preprocessing [19]. Figure 12 shows that the large EMG spikes are only 'rounded-off'. They cannot be removed as their power is smeared over a wide frequency band. As a consequence, their frequency spectrum overlaps that of EDA. Due to this effect, the peaks are very hard to remove using static frequency filters alone.

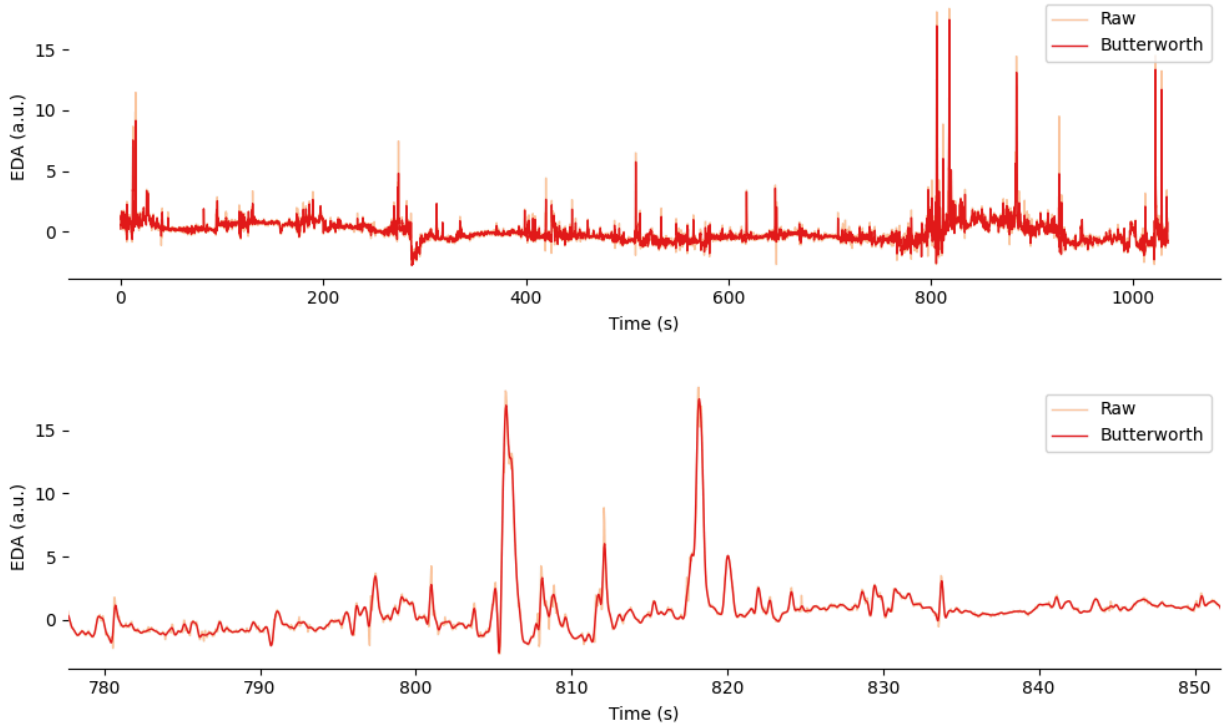

Figure 12: EDA as recorded by the IM-TWIN T-Shirt in its raw and filtered form. The raw EDA was filtered using a 4-th order low-pass Butterworth filter having a cutoff frequency of 3 Hz. Clearly, the low-pass filter is not capable of removing the high-amplitude spikes as their power is smeared over a large frequency band.

As such, we address the problem from a different angle. Rather than employing a single coarse low-pass filter that eliminates the entire frequency spectrum above a specific frequency (e.g., cutoff frequency), we opt for multiple bandpass filters where each filter aims to identify a particular power threshold. This technique, inspired by [19], leverages this adaptive power threshold to remove strong frequency components atypical in EDA signals. The EDA signal is then reconstructed using the filtered frequency components.

In our processing pipeline, the multiple bandpass filters are realized through a Stationary Wavelet Transform (SWT) [21], a method akin to the Discrete Wavelet Transform (DWT) [22]. Both SWT and DWT utilize cascades of bandpass filters to break down a signal into short atoms known as 'wavelets' that are localized in time and frequency. DWTs are streamlined for speed and crafted for mathematical perfection in reconstruction (e.g., no redundancy) [22]. Therefore, they only use frequencies that are powers of 2 to dissect a signal, and the signal is downsampled by a factor of 2 following each wavelet transformation. This dyadic process of decomposition is well-suited for compression, but less effective for processing as it distorts the time-domain [23]. The SWT addresses this by preserving the signal length after each wavelet, while utilizing a dyadic frequency scale. Figure 13 illustrates the SWT decomposition of an EDA

signal into four levels using the Haar wavelet. Note that the dyadic frequency scale ensures there is no overlap between the different levels.

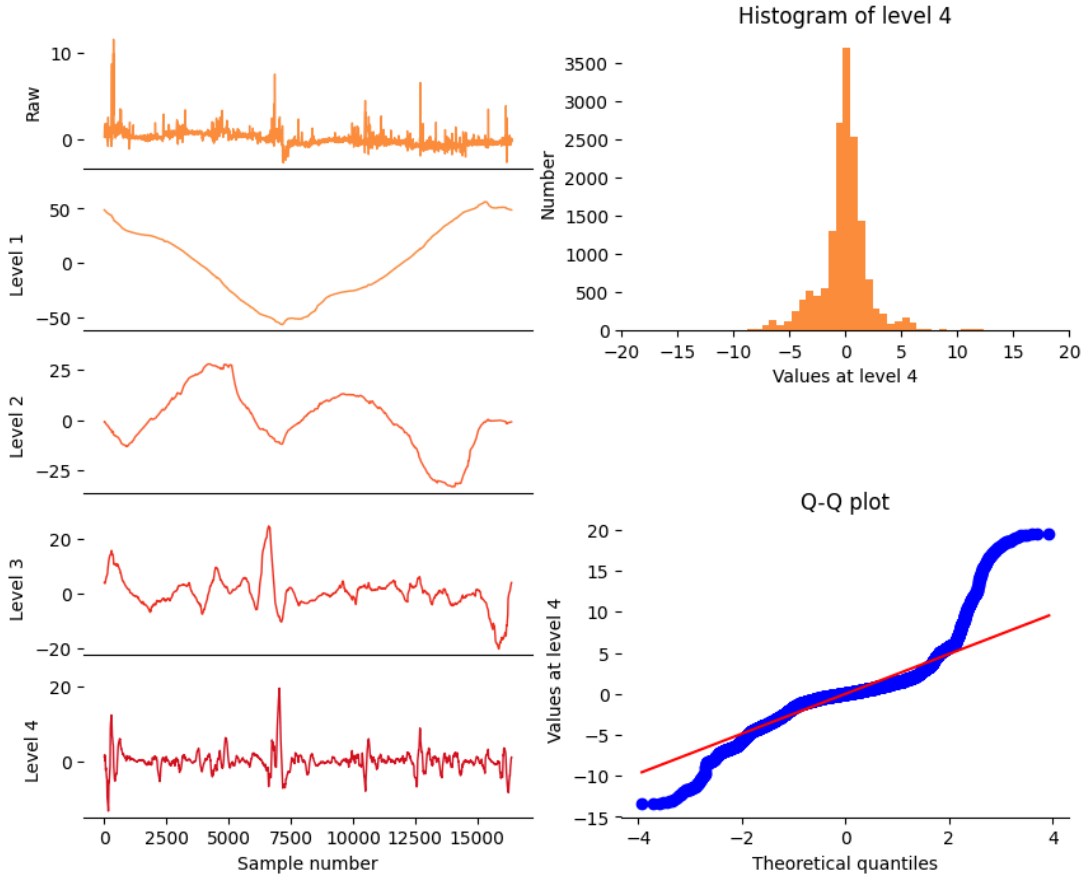

Figure 13: The raw EDA signal and its wavelet coefficient components at level 1-4. Top right: The distribution of coefficient values at level 4. Bottom right: QQ-plot of the same distribution showing a clear deviance from normality at 2 SD's from the mean or above a value of 5.

An important characteristic of the EDA signal is that power is normally distributed across each frequency band [19]. In other words, all wavelet coefficients at a given level follow a normal distribution. Figure 13 illustrates this effect for the wavelet coefficients of level 4. Since we expect the coefficients to conform to a normal distribution, high-amplitude noise can be readily identified as statistical outliers (as seen in the QQ plot of Figure 13). Within the [-2,2] quantile range, the distribution adheres to the QQ-line. However, outside this region it deviates significantly. Utilizing this method, we can establish a threshold for each wavelet level, to confidently mark wavelet coefficients as noise or signal.

However, setting the threshold solely based on the QQ-plot would not be fully correct. A more precise representation of the wavelet coefficients in an EDA signal involves the superposition of two Gaussian distributions [19]. One that resembles the changes in SCL having a large variance, while the other represents the SCR, characterized by a smaller variance but a higher

peak. To sustain unit normalization (e.g., ensuring the area under the curve equals 1), the two distributions are scaled by variables g and (1−g), respectively. Additionally, the second Gaussian's standard deviation equals the first Gaussian's standard deviation, multiplied by a factor c. The function can be described as follows:

$$f(x) = \frac{g}{\sqrt{2\pi s^2}} \cdot e^{-\frac{x^2}{2s^2}} + \frac{1-g}{\sqrt{2\pi (c \cdot s)^2}} \cdot e^{-\frac{x^2}{2(c \cdot s)^2}},$$

where g, s and c are variables that are used to fit f(x) to the wavelet coefficient distribution. The function is fitted to each wavelet level individually. Subsequently, the cumulative probability density function (CDF) is computed, facilitating the identification of thresholds d and −d, between which (1−p) percent of the data samples fall. The value for p is empirically set to 0.01, meaning that the minimum and maximum 0.5% of the data is considered as noise.

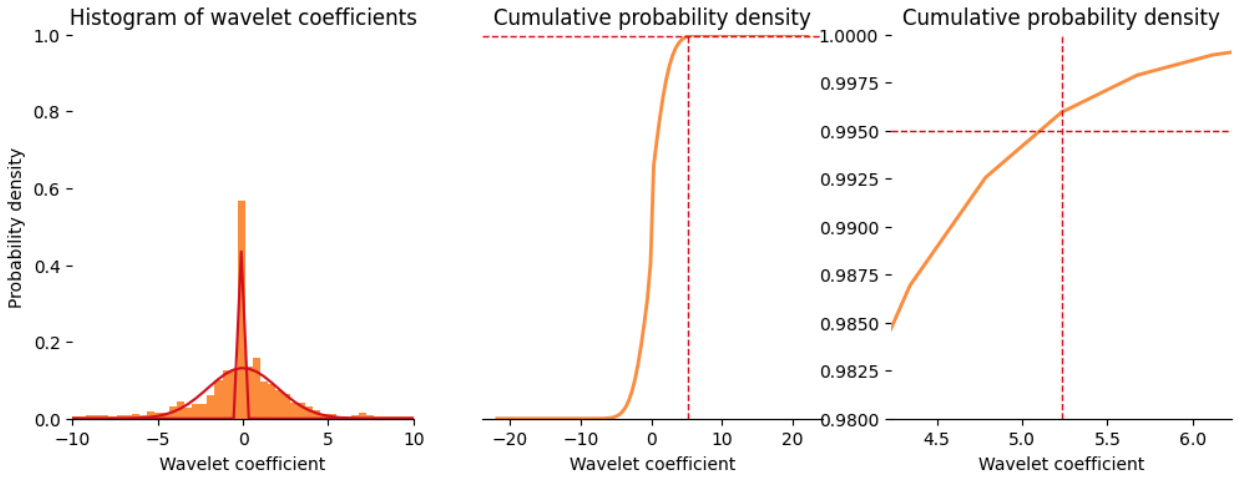

Figure 15: A superposition of two Gaussian functions is fitted to each wavelet coefficient distribution. Subsequently, the Cumulative Probability Densitiy (CDF) function is calculated using integration. This allows us to retrieve the threshold d for which 99% of the data lies between [-d,d]. Noise is marked as data samples falling outside this range.

Wavelet coefficients that fall outside the [-d,d] region are set to zero. This process repeats for all wavelet levels between [0.05-1.0] Hz. Because the majority of EDA's phasic power lies in the [0.05-1.0] Hz range [20], wavelet coefficients in levels above 1 Hz are most likely to be noise and are, hence, set to zero. Similarly, wavelet coefficients in the levels below 0.05 Hz belong to the tonic component and are not filtered. The result is a selectively low-pass filtered EDA signal (see Figure 16).

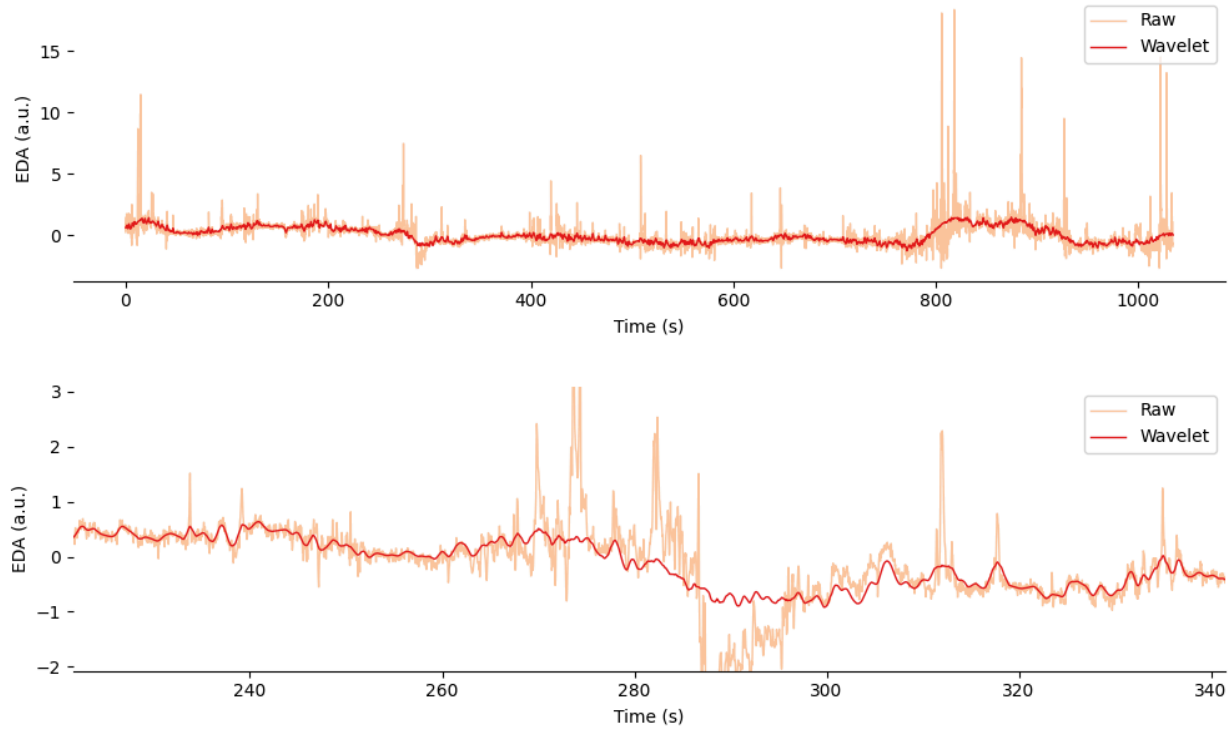

Figure 16: Raw and Wavelet filtered EDA signal. In contrast to the Butterworth approach in Figure 12, the Wavelet filtering approach achieves much better results due to its selective filtering in the EDA frequency band.

Figure 16 illustrates the enhanced ability of the by Tronstad et al. [19] inspired adaptive wavelet filter to eliminate high-amplitude EMG noise, particularly when compared to the low-pass Butterworth filter. The bottom plot of Figure 16 also reveals the filter's capability in eradicating excessive low-frequency baseline wander, while still effectively discerning the skin conductance level at the start of the recording. Consequently, the proposed wavelet filter genuinely enhances signal quality, laying a solid foundation for subsequent processing.

2.3.3 EDA feature extraction

Following the filtering, we continue with the feature extraction phase, starting by dividing the EDA into its phasic and tonic components. The enhanced signal quality markedly improves usability of the phasic component and visibly betters the tonic component as well. Figure 17 shows an example of the separation of the phasic and tonic elements of a wavelet-filtered EDA signal.

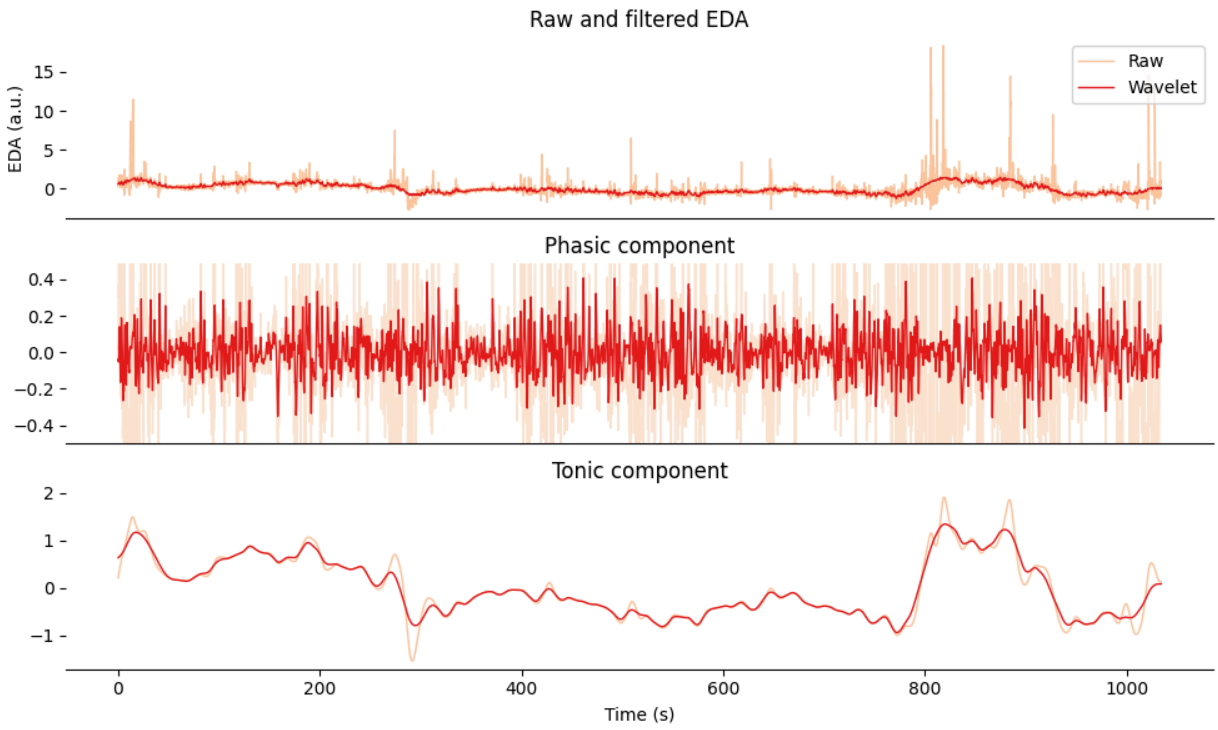

Figure 17: The first step in EDA feature extraction is the separation of the phasic and tonic components. The plot shows the separation for both the raw and wavelet filtered EDA signals. The phasic component is especially improved by the wavelet filter.

The tonic component can be utilized without additional processing. Analogous to the ECG feature extraction method, the tonic signal can be segmented using a trailing 120-second window. From there, moment-based features such as the mean, standard deviation, skewness, and kurtosis of the signal can be extracted.

The phasic component, however, presents more of a challenge. Normally, phasic analysis would start with peak detection [12]. These peaks, representing specific or nonspecific skin conductance responses (S-SCRs or NS-SCRs), would then be analyzed for attributes like height, duration, halftime, etc [17]. Unfortunately, even after wavelet denoising, the EDA signal remains substantially saturated in the [0.04-0.4] Hz region; the band where NS-SCRs are found (see Figure 17). Consequently, traditional threshold-based peak detection algorithms are unsuitable (see Figure 18), as the noise causes the signal to surpass the threshold for peak detection far more frequently than would be the case with a clean EDA.

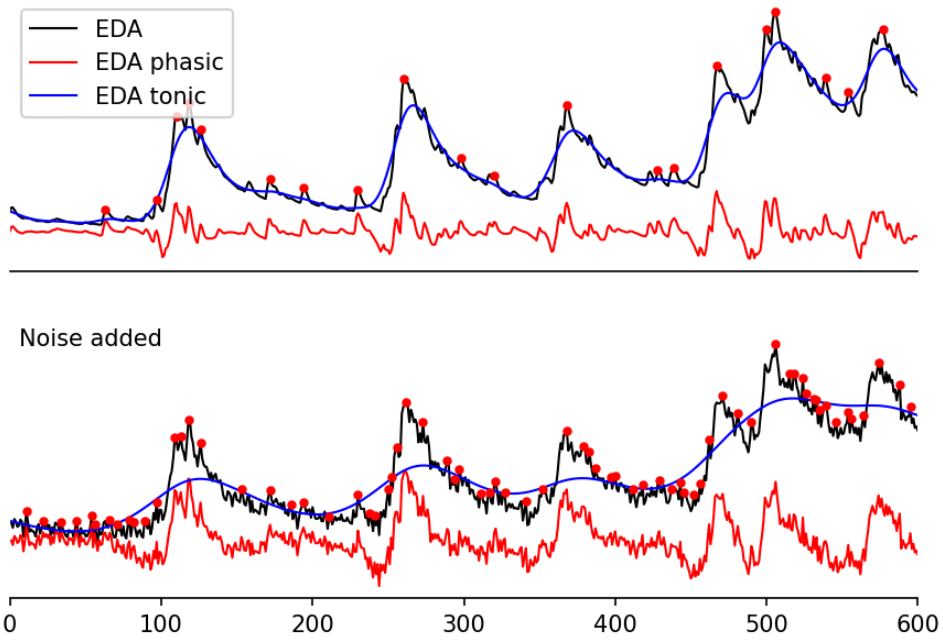

Figure 18: An example EDA signal with and without White Gaussian Noise (WGN) added. In both cases, the EDA is decomposed into its phasic and tonic components. The phasic component is searched for peaks to extract NS-SCRs. The added noise distorts this process severely.

To solve this problem, we propose a feature based on the wavelet transformation. While we utilized the Stationary Wavelet Transform (SWT) for signal decomposition and reconstruction, we employ a different variant for feature extraction – the Continuous Wavelet Transform (CWT). Extensively detailed in report D2.2, we leverage our fast implementation of the CWT, referred to as fCWT [23], to compute a comprehensive time-frequency representation.

Inspired by previous research [20], we then extract the mean frequency power within the range of 0.04 to 0.4 Hz over time. This resulting feature, termed 'TF mean', is subsequently segmented using the trailing window technique, a method we also applied for HR, HRV, and tonic feature extraction.

Figure 18 illustrates both the TF mean and a conventional phasic peak-based feature, known as the SCR count, which measures the number of peaks within a particular time frame. The high correlation between these two features validates the use of TF mean as a phasic feature. As noise is automatically mitigated by the wavelet transform [23], the approach manages to effectively represent the phasic behavior of the EDA signal, even in the presence of significant noise.

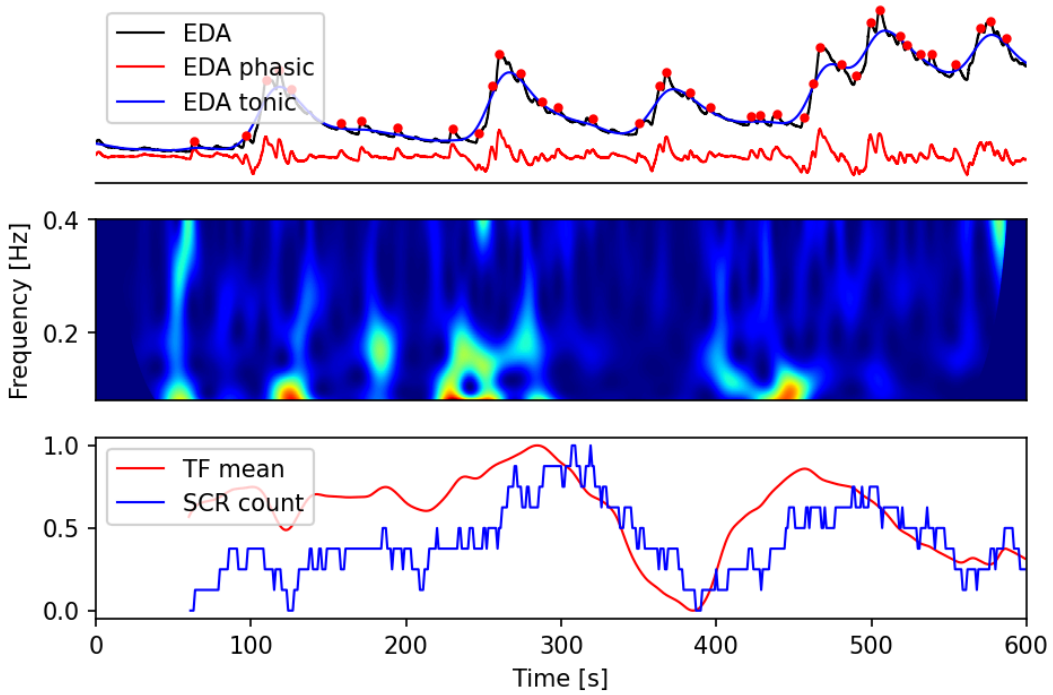

Figure 19: We designed a time-frequency based feature to describe phasic activity in noisy EDA signals. In this figure we compare the mean power in a [0.045-0.4] Hz frequency band to a renowned phasic feature; the number of peaks in a certain time-frame, SCR count.

To demonstrate the reliability of the newly developed feature, we conducted an experiment to test its resilience to noise in comparison to the traditional SCR count feature. We progressively contaminated a clean EDA with noise. Subsequently, we calculated the correlation between the feature extracted from both the clean and noisy EDA signals at each noise level, and plotted the result in Figure 20.

As anticipated, the peak-based SCR count showed significant deterioration at noise levels even as low as 24dB. In contrast, the TF mean feature exhibited robustness, maintaining decent correlation even at 12dB noise level. As such, the TF mean feature shows promise to be applied in the IM-TWIN system.

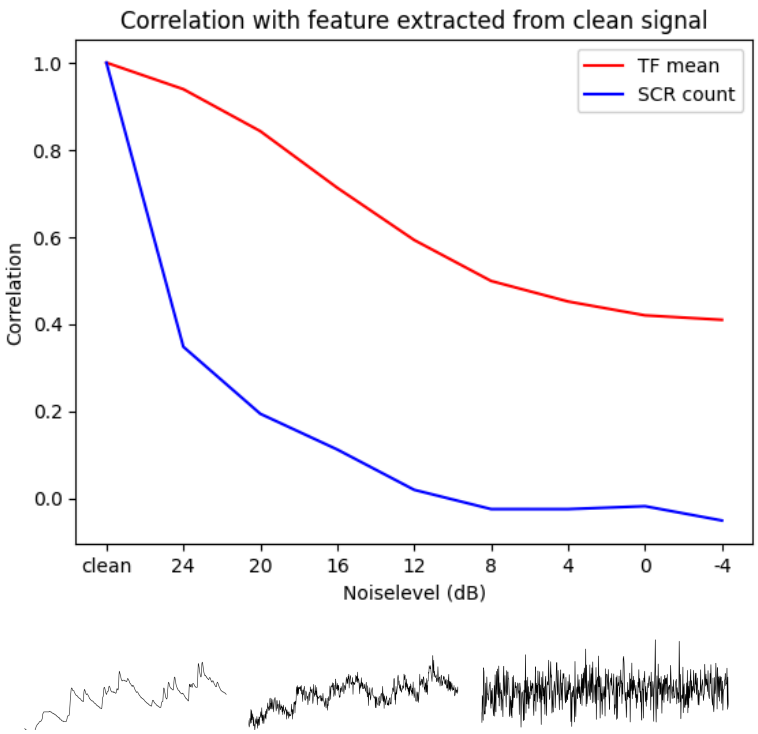

Figure 20: Noise-resilience plotted as the correlation between a feature extracted from the clean and noisy signal for different levels of added noise. As expected, the SCR count feature is severely impacted even by low levels of noise as it relies on fixed threshold peak extraction.

Together with the tonic component, the TF mean feature extracted from the phasic component is segmented into 120s trailing windows. Next, the signals are described by statistical moment-based features such as the mean, standard deviation, skewness and kurtosis. Additionally, other features based on information theory such as approximate or sample entropy could be extracted. See report D3.2 for specific information about the final stage of feature extraction and classification.

3. Processing of visual information

In the previous work (see deliverables[1] D2.1 and D3.3), CNR-ISTC described a Python implementation of a software, designed to detect the eye contact between child and therapist. The "*eye contact detector tool*"[2] processes the video recorded through *camera glasses* worn by

[1] D2.1 *Processing of physiological signals, visual information, and PlusMe interaction: first version*, and D3.3 "*Plusme AI-augmente behavior and IM-TWIN 1*", available at https://im-twin.eu/deliverables/
[2] See video https://im-twin.eu/video/#eye_contact_detector

the caregiver during a play session with the child, and produces as output a log file scoring the number of eye contact events (see Figure 21).

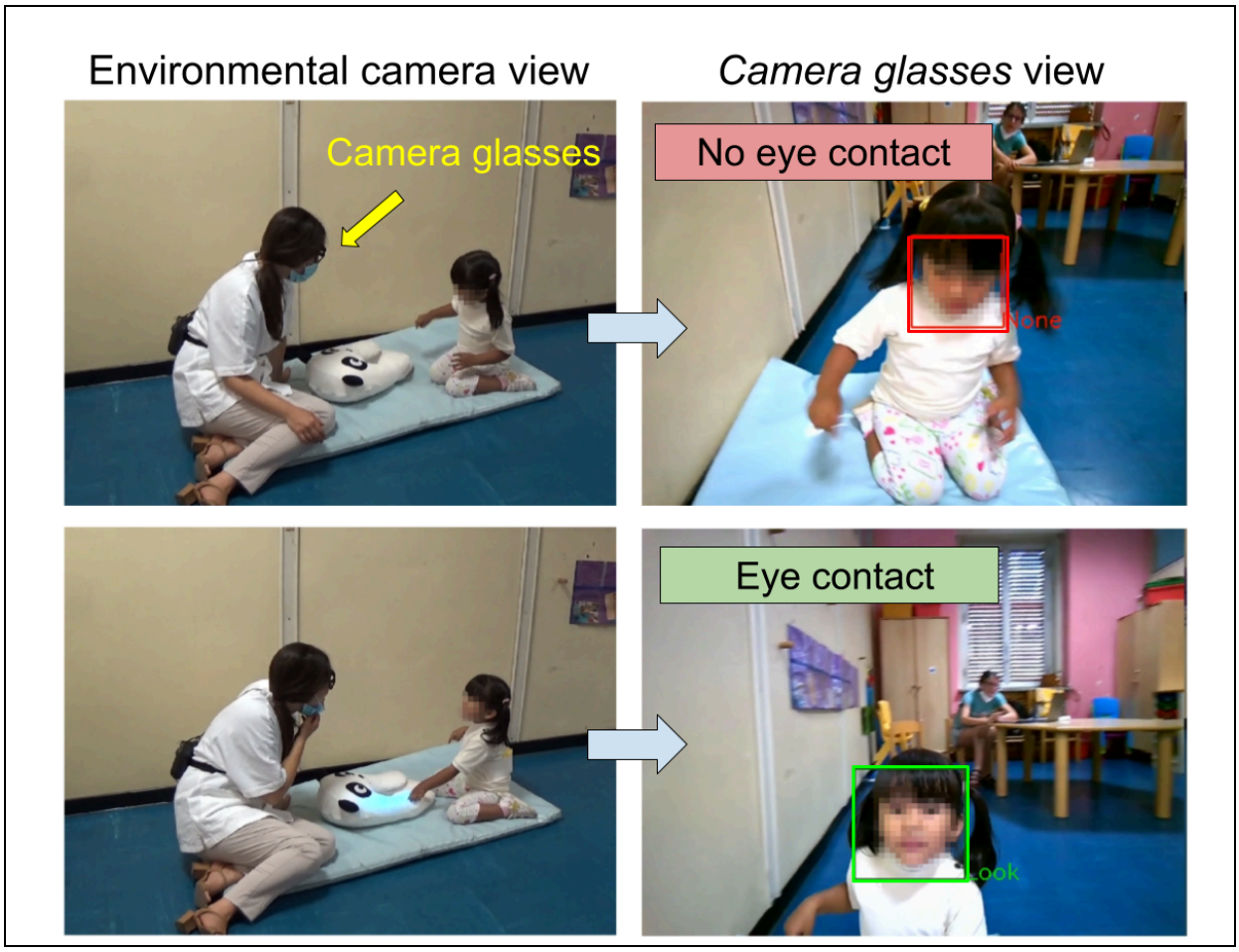

Figure 21: a pilot test run at SAPIENZA, where the "Eye contact detector tool" was used to detect the eye contact events between child and therapist.

In order to make this tool more accessible and usable by non-expert experimenters, CNR-ISTC implemented a Graphic User Interface (GUI) to manage the software. Through the GUI the user can upload a video, previously captured by the *camera glasses* during the experimental session, select the main parameters relevant for the analysis, and process the data.

A video about the application is available in the project website, at the following link: https://im-twin.eu/video/#linux_app_eye_contact_detector.

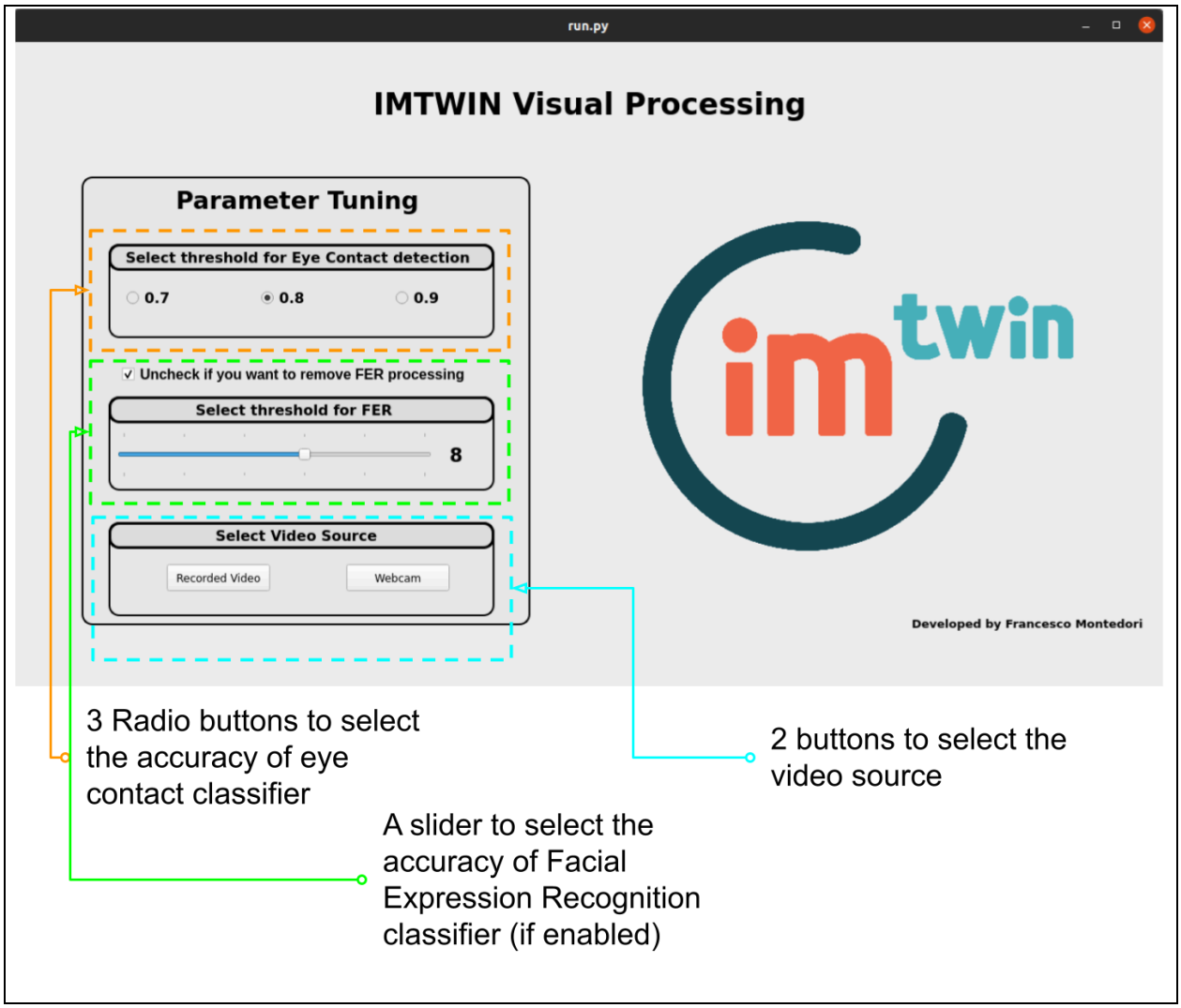

Figure 22: the GUI of the "Eye contact detector tool".

In detail, the GUI (see Figure 22) allows the user to:

- select the threshold for eye contact detection. This is a percentage which determines the accuracy of the classifier (which relies on a recent computer vision algorithm by Chong and colleagues[3]). Three values are available: 70%, 80%, 90%. In the current pilot tests at SAPIENZA, the value 80% is used, as it provides the most reliable results when compared with the response of a human rater (Inter Rater Reliability, k≈0.80).
- select the video source: "Recorded Videos" or "Webcam". In case the "Recorded Video" button is pressed, a popup window shows up asking the user to select the file to

[3] Chong, E., Clark-Whitney, E., Southerland, A. *et al.* Detection of eye contact with deep neural networks is as accurate as human experts. *Nat Commun* **11**, 6386 (2020), https://doi.org/10.1038/s41467-020-19712-x

process. In case the “Webcam” button is pressed, the popup window asks the user to verify the USB camera connection. This option can be used for debug purposes, and it allows to check the performance of the application in real time.

When the video source is selected, the script of the application loads the model weights and the processing of video begins (Figure 23).

Figure 23: during the processing, the application provides the user with feedback about the performance of the eye contact classifier (in this example, also the output of the Facial Expression Recognition classifier is reported).

During the processing, the application provides the user with an immediate feedback about the performance of the eye contact classifier, by streaming the original video and overlaying on the image the detected face frame, the eye contact response (“Look” / “No Look”), and the confidence percentage; moreover, a countdown label informs the user about the estimated time until the end of processing[4].

Once the processing is terminated, the application creates a folder containing 2 files which report the classifier outputs:

- a txt file: this is a log file where, for each frame, is provided the response (0 or 1) of the eye contact classifier, according to the threshold accuracy selected;

[4] The speed of processing depends on the video resolution: 640*480 is processed quite in real-time (about 24 fps), while 1980x1080 is processed at about 1 fps.

- an mp4 file: this is the feedback video shown to the user during the processing. The video is useful for debug purposes, when the user wants to compare the classifier performance with the results in the txt log file.

It is important to underline that the frame rate of the video input is always downsampled[5] at 20 fps before the processing, so that the related log file presents 20 readings per second. This resampling is necessary in order to make the sampling rate of the application compliant with the sampling rate of the TWC toys, which collect data at 20 Hz. The reason for this compliance is explained in the next section 4 "*Processing of interaction between child, PlusMe and therapist*".

Although the primary purpose of the application is to assess the eye-contact, the GUI also allows the user to include in the visual processing, as an additional – still experimental – feature, the classification of the Facial Expression Recognition (FER). This processing was part of the original IM-TWIN research plan but, as shown in a previous deliverable D2.1[6], the FER module performed poorly in the real scenario with children, and was temporarily set aside. To overcome this limitation, CNR-ISTC reimplemented then the FER algorithm, using the POSTER++ model, recently published by MAO et al. in 2023[7]; this is an improvement of the previous version of the POSTER model, which achieves the state-of-the-art performance in FER, but featuring an undoubtedly complex architecture and requiring expensive computational costs. The new model POSTER++, with the minimum computational cost, reaches an overall accuracy of 92.21% on RAF-DB[8], a reference dataset for face expressions (see Figure 24).

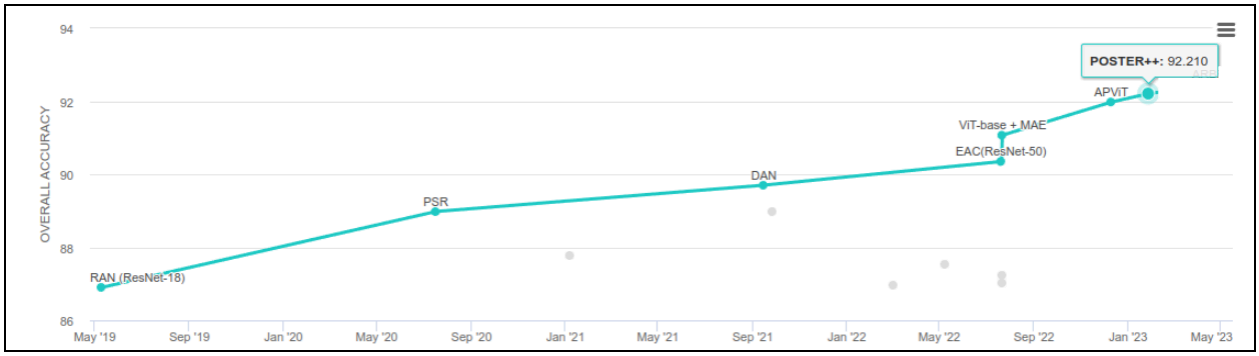

Figure 24: the performance of POSTER++ model in facial expression recognition task.

To test the new model, the GUI of the application (see Figure 22) allows the user to activate, along with the eye-contact detection, the FER processing, also selecting the accuracy threshold. In this case the application outputs (the txt log file and the mp4 video file) also report the FER results (see Figure 23).

[5] Standard cameras generally provide a video recording of 30 or 25 fps.
[6] D2.1 *Processing of physiological signals, visual information, and PlusMe interaction: first version*, available at https://im-twin.eu/deliverables/
[7] J. Mao, R. Xu, X. Yin, Y. Chang, B. Nie, A. Huang, *POSTER++: A simpler and stronger facial expression recognition network* (2023), https://arxiv.org/abs/2301.12149
[8] See http://www.whdeng.cn/raf/model1.html for further information.

The FER performance with ASD children is currently in evaluation, during the experimental sessions run at SAPIENZA facility.

As next step, to facilitate access and use of the GUI by researchers, the CNR-ISTC plans to embed the software in a Linux AppImage[9]: this is a kind of single, executable file for Linux systems, which contains all the dependencies to run the application as a stand-alone component. The AppImage, after IPR evaluation, could be made freely available in the project GitHub page: https://github.com/IM-TWIN.

4. Processing of interaction between child *PlusMe* and therapist

In the previous work[10] CNR-ISTC described the first implementation of the TWC software to collect the data about the physical manipulation of the toy – *Octopus X-8*, in the feasibility test – by the child. In detail, the software showed how the combination of data provided by a TWC and the *camera glasses* worn by the therapist, can provide interesting information about the child's social behavior during the play activities (see Figure 25).

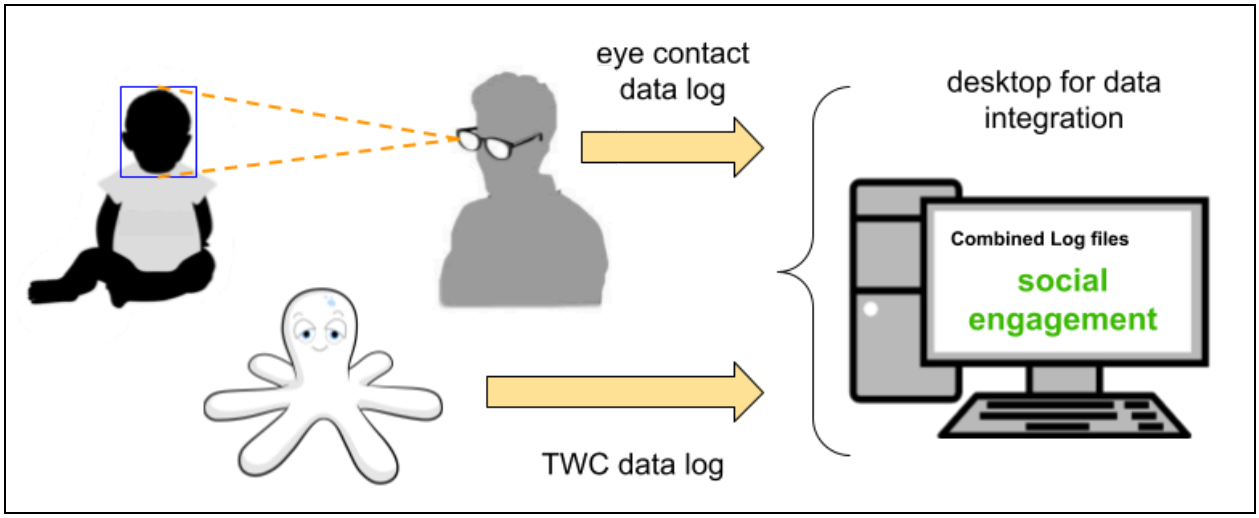

Figure 25: schema for data integration, combining the input from the sensorised glasses and the TWC toy (adapted from previous deliverable 2.1).

In detail, as shown in Figure 26, both tools provide a log file[11]:

[9] See https://appimage.org/ for further details.
[10] D2.1 *Processing of physiological signals, visual information, and PlusMe interaction: first version*, available at https://im-twin.eu/deliverables/
[11] *A video of the X-8 data collection capabilities, including the sensorised glasses, is available at the project webpage link* https://im-twin.eu/video/#x8_functional_features.

- the *camera glasses* produces a log file (sampling rate: 20 Hz) with the evaluation of eye contact between child and therapist;
- the *Octopus X-8* – the TWC used in the feasibility test – produces a log file (sampling rate: 20 Hz) with several data such as:
    - status of the *X-8* tentacles activations (touched / not touched), including the identity of the user who touches the toy (child / caregiver);
    - triggering of toy sensory reward (lights and sounds), including the type of reward;
    - type of game, currently selected by the caregiver (in *X-8,* 3 different games with increasing complexity are available, while Panda *PlusMe* presents 6 games);
    - if the rule of the selected game requires a turn, the specification of the game round (e.g.: 0: game with no turn; 1: child's turn; 2: caregiver's turn).

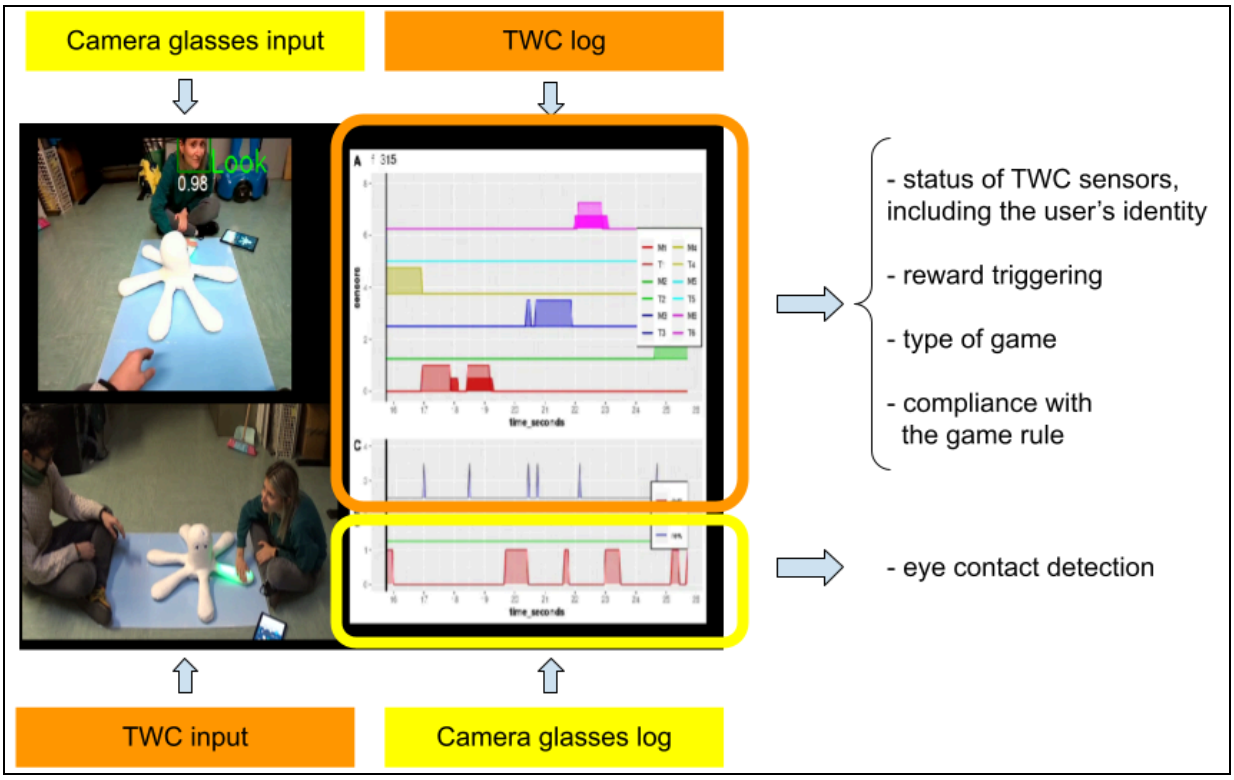

Figure 26: the 2 logs recorded by the camera glasses and the TWC Octopus X-8. The integrated data provides interesting information about the social interaction between child and therapist. In this example the logs have been synchronized by hands.

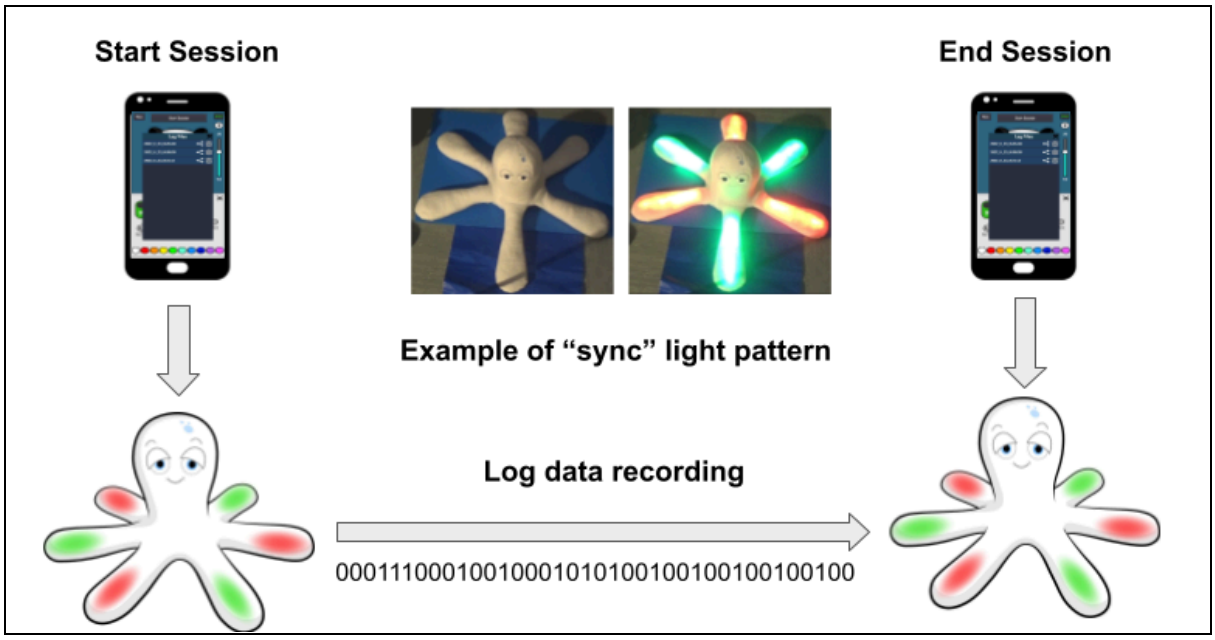

Figure 27: the TWC device produces a brief light pattern when the recording of data log starts or stops. This signal is necessary to sync the log file with the video recording of the sessions.

It is important to note that the fact that the two devices are activated independently at different points in time can cause possible synchronization issues. In other words, once the logs are available at the end of the session, the researcher has still to sync *by hand* the two files, so that the recorded events collected by the TWC log temporally match with those collected by the *camera glasses* log. This manual task, can be tricky for the following reasons:

- the operation requires a software for video editing and the competence for using it. Through the software, the researcher has to pair the videos from the environmental camera and the *camera glasses*, and synchronize them. This is done exploiting the "sync" light pattern, namely a brief light signal (about 1 second) produced by the toy when the log session starts / stops (see figure 27). This visual mark is used as a "clapper board" to pair the videos from both cameras;
- if the therapist wearing the *camera glasses* misses the TWC synch light signal[12], the pairing of videos can become even more difficult.

In order to overcome these possible obstacles (which could prevent the actual use of the tool by researchers), CNR-ISTC started to develop a new implementation of the *camera glasses*. In more detail, the original cam[13] embedded in the glasses has been substituted with the "*Camera sensor module "XIAO ESP32S3 Sense"*[14]. This module supports 2.4 GHz WiFi and BLE dual

[12] This can happen if the therapist is not looking at the TWC when the synch light is produced.
[13] endoscope camera module, model "CMT-8MP-IMX179-W510 USB", by http://camera-module.com/
[14] https://www.seeedstudio.com/XIAO-ESP32S3-Sense-p-5639.html?queryID=c81515975f93149921afaa5548ee8b43&objectID=5639&indexName=bazaar_retailer_products

wireless communication, while the camera provides a 1600 x 1200 resolution, with a 68.7° field of view (see Figure 28).

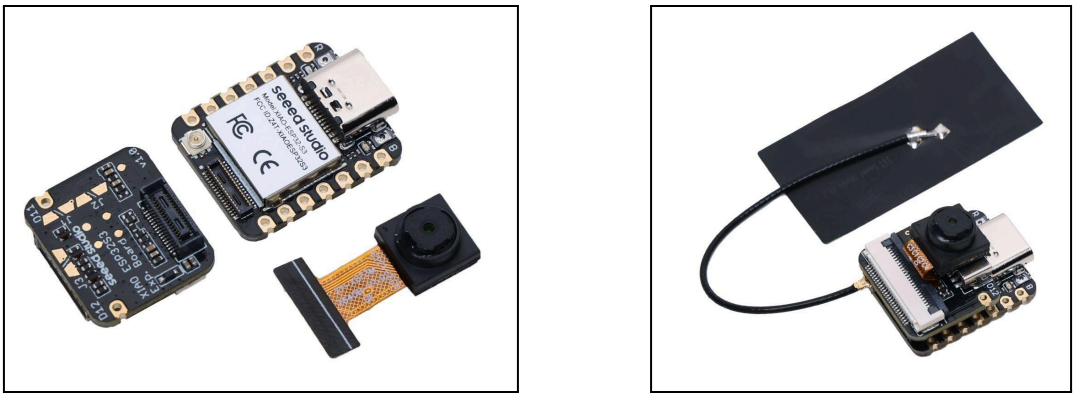

Figure 28: the camera sensor module XIAO ESP32S3 Sense.

This module features a very interesting attribute for the IM-TWIN system components: it is based on the ESP32 board, the same microcontroller used in the *Panda PlusMe* and *Octopus X-8*. This allows the TWC Android app to control both the selected toy and the camera module embedded in the *camera glasses* (see Figure 29), and to manage the automatic synchronization of the TWC log file and camera recording, by making the two processes start at the same time.

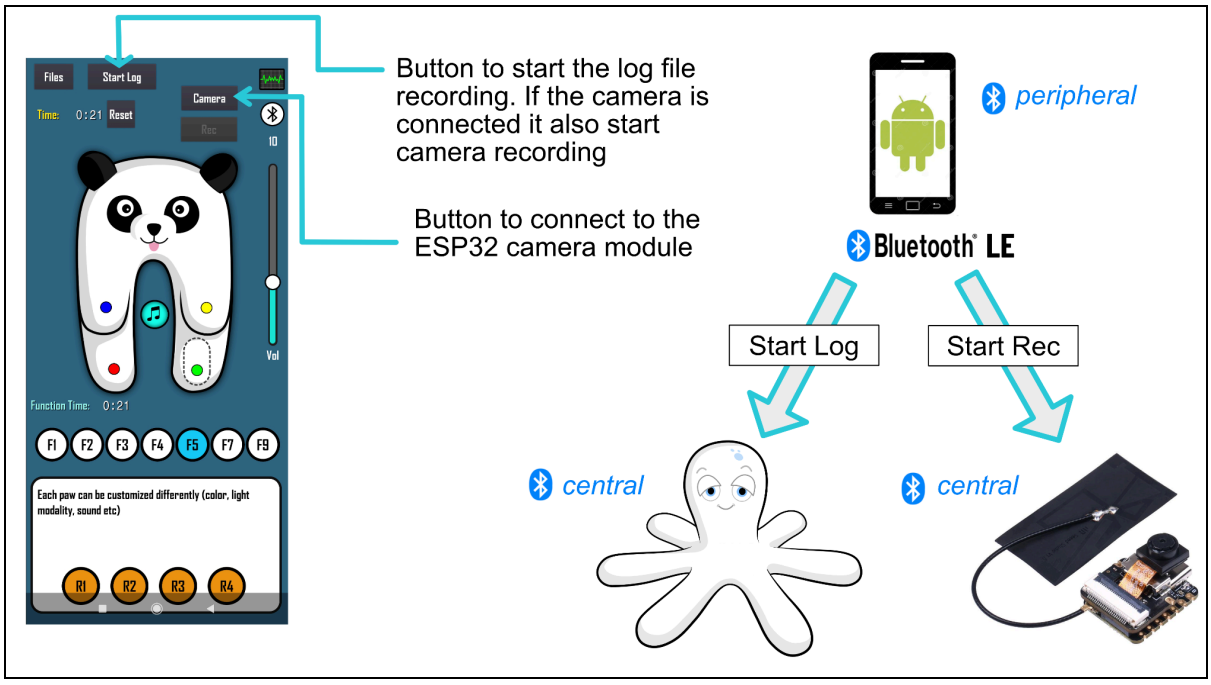

Figure 29: the improved mobile Android app can now control both the TWC toy and the camera module embedded in the camera glasses.

From a technical point of view, this is achieved by ensuring that all devices are properly configured according to the Bluetooth Low Energy protocol (BLE). BLE devices operate in two primary roles, *peripheral* and *central*. *Peripherals* broadcast data or services, while *centrals* scan for and connect to these *peripherals*. One *peripheral* can connect to several *centrals* concurrently; one *central* can connect to only one *peripheral*. In our case the mobile Android application is configured as a *peripheral* while the TWC and camera module are configured as *centrals* (see Figure 30).

The TWC application was then modified as follows:

- a new “Camera” button has been added in the GUI (see Figure 29). The button checks if the camera module is available; this happens when the new *camera glasses* – now based on ESP32 – are activated and within the bluetooth working range;
- if the camera module is found, the application connects to it and the camera button turns blue, indicating that a connection has been established;
- when the researcher presses the “Start Log” button to start the TWC data collection (see Figure 26), the application sends a command to the camera module, which starts the video recording; conversely, when the researcher stops the TWC log file, the command also stops the video recording. As a consequence the TWC log file “matches” temporally with the video collected by the *camera glasses*.

This hardware/software improvement simplifies the procedure of data collection and analysis, which can be summarized in the following steps:

- once the experimental session is terminated, the researcher saves the data log produced by the TWC;
- the researcher then processes the video recorded by the *camera glasses*, using the GUI of the “*eye contact detector tool”*, described in the previous section 3 “Processing of visual information”. The processing produce the second data log, concerning the eye contact between child and researcher;
- the researcher, using a standard statistical tool (e.g. “R”), can join the two logs, with no longer need to pre process the visual data through a video editor. Both logs now report the same events, temporally paired, sampled at 20 Hz.

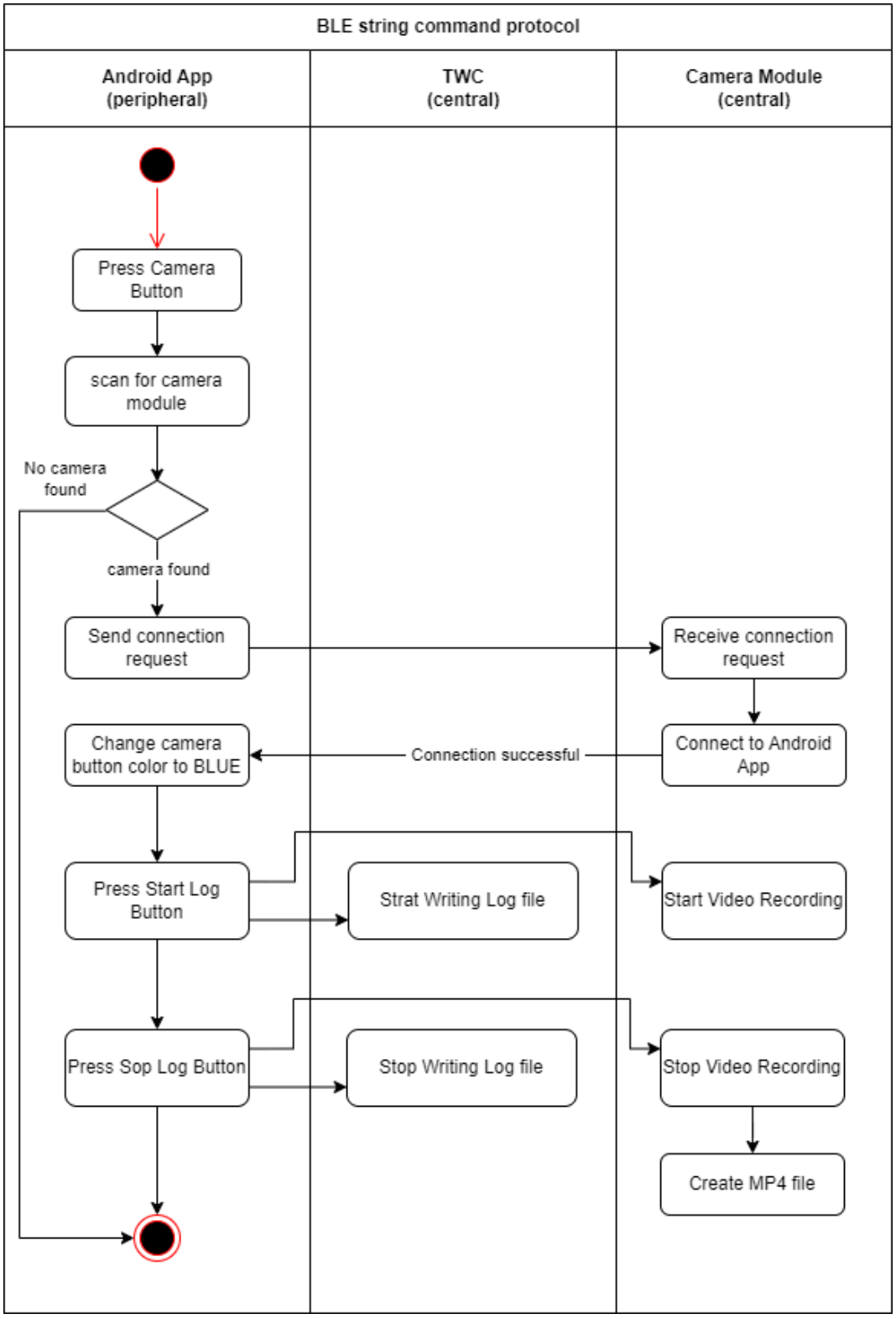

Fig. 30: UML Activity Diagram showing how the log file on the TWC and the video recording from the camera module, synchronize when the camera start log button is pressed.

5. Conclusion and future developments

In the current study, we have introduced a processing pipeline tailored for the IM-TWIN system, designed to filter out extraneous variance in both physiological and video signal data. As this approach reduces extraneous variability, it allows for the deployment of simpler, more streamlined models during the classification phase, detailed further in Report D3.2.

Looking forward, there are opportunities to evolve the physiological processing pipeline towards real-time operation. Given that the existing system already demonstrates computational efficiency, the primary obstacle lies in algorithmic refinement. As it stands, the pipeline operates on complete data records. To adapt to real-time demands, a transition to a streaming input mechanism—capable of processing smaller, successive data segments—would be essential.

Concerning the processing of data collected by TWCs toys and by *camera glasses,* CNR-ISTC presented two improvements of the software which manage the logs of the two devices. Such enhancement was developed to facilitate and promote the use of the tools by non-expert researchers. In detail CNR-ISTC presented a GUI for the "*eye contact detector*" tool (see sec. 3 "Processing of visual information"), and a new components integration which solve the problem of data synchronization from different devices, namely the TWC toys and the *camera glasses* (see sec. 4 "Processing of interaction between child, PlusMe and Therapist"). Both improvements are currently (September 2023) in test phase at SAPIENZA.

Finally, effort has to be made on fusing the two datastreams. Currently, two separate models are trained on the physiological and camera features. After the testing phase, future efforts could be focused on bringing these features together in one model. Combining the two sources of information could leverage the performance of the whole system.

6. References

1. Rajpurkar, P., Chen, E., Banerjee, O. et al. AI in health and medicine. Nat Med 28, 31–38 (2022). https://doi.org/10.1038/s41591-021-01614-0
2. Guarding against the uncertain perils of AI. Nat. Biomed. Eng 7, 705–706 (2023). https://doi.org/10.1038/s41551-023-01064-8
3. van den Broek, E. (2011). Affective Signal Processing (ASP): Unraveling the mystery of emotions. PhD-thesis, Human Media Interaction (HMI), Faculty of Electrical Engineering, Mathematics, and Computer Science, University of Twente, Enschede, The Netherlands. ISBN: 978-90-365-3243-3. DOI: https://doi.org/10.3990/1.9789036532433 .
4. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

5. Begoli, E., Bhattacharya, T. & Kusnezov, D. The need for uncertainty quantification in machine-assisted medical decision making. Nat Mach Intell 1, 20–23 (2019). https://doi.org/10.1038/s42256-018-0004-1
6. Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, & Nikolai Liubimov (2020-2022). Label Studio: Data labeling software.
7. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
8. Hannun, A.Y., Rajpurkar, P., Haghpanahi, M. et al. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nat Med 25, 65–69 (2019). https://doi.org/10.1038/s41591-018-0268-3
9. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.
10. Kramer, M. A. (1991). Nonlinear principal component analysis using autoassociative neural networks. AIChE journal, 37(2), 233-243.
11. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
12. Cacioppo, J. T., Tassinary, L. G., & Berntson, G. (Eds.). (2007). Handbook of psychophysiology. Cambridge university press.
13. Camm, A. J., Malik, M., Bigger, J. T., Breithardt, G., Cerutti, S., Cohen, R. J., ... & Singer, D. H. (1996). Heart rate variability: standards of measurement, physiological interpretation and clinical use. Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology. Circulation, 93(5), 1043-1065.
14. Pan, J., & Tompkins, W. J. (1985). A real-time QRS detection algorithm. IEEE transactions on biomedical engineering, (3), 230-236.
15. Porr, B., & Howell, L. (2019). R-peak detector stress test with a new noisy ECG database reveals significant performance differences amongst popular detectors. BioRxiv, 722397.
16. Zahid, M. U., Kiranyaz, S., Ince, T., Devecioglu, O. C., Chowdhury, M. E., Khandakar, A., ... & Gabbouj, M. (2021). Robust R-peak detection in low-quality holter ECGs using 1D convolutional neural network. IEEE Transactions on Biomedical Engineering, 69(1), 119-128.
17. Boucsein, W. (2012). Electrodermal activity. Springer Science & Business Media.
18. Healey, J., & Picard, R. (1998, May). Digital processing of affective signals. In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181) (Vol. 6, pp. 3749-3752). IEEE.
19. Tronstad, C., Staal, O. M., Sælid, S., & Martinsen, Ø. G. (2015, August). Model-based filtering for artifact and noise suppression with state estimation for electrodermal activity measurements in real time. In 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 2750-2753). IEEE.
20. Posada-Quintero, H. F., Florian, J. P., Orjuela-Cañón, Á. D., & Chon, K. H. (2016). Highly sensitive index of sympathetic activity based on time-frequency spectral analysis of electrodermal activity. American Journal of Physiology-Regulatory, Integrative and Comparative Physiology, 311(3), R582-R591.
21. Nason, G. P., & Silverman, B. W. (1995). The stationary wavelet transform and some statistical applications. In Wavelets and statistics (pp. 281-299). New York, NY: Springer New York.
22. Shensa, M. J. (1992). The discrete wavelet transform: wedding the a trous and Mallat algorithms. IEEE Transactions on signal processing, 40(10), 2464-2482.
23. Arts, L. P., & van den Broek, E. L. (2022). The fast continuous wavelet transformation (fCWT) for real-time, high-quality, noise-resistant time–frequency analysis. Nature Computational Science, 2(1), 47-58. https://www.nature.com/articles/s43588-021-00183-z

24. Laborde, S., Mosley, E., & Thayer, J. F. (2017). Heart rate variability and cardiac vagal tone in psychophysiological research–recommendations for experiment planning, data analysis, and data reporting. Frontiers in psychology, 8, 213.
25. Kemper, K., Hamilton, C. & Atkinson, M. Heart Rate Variability: Impact of Differences in Outlier Identification and Management Strategies on Common Measures in Three Clinical Populations. Pediatr Res 62, 337–342 (2007). https://doi.org/10.1203/PDR.0b013e318123fbcc
26. Cajal D, Hernando D, Lázaro J, Laguna P, Gil E, Bailón R. Effects of Missing Data on Heart Rate Variability Metrics. Sensors (Basel). 2022 Aug 2;22(15):5774. doi: 10.3390/s22155774. PMID: 35957328; PMCID: PMC9371086.

History of Changes

No.	Description
1	Version updated from 1 to 2 (March 2024)
2	At the end of section 2.1 “Signal Quality Indicator” (page 6), the following content was added: “*It should be noted that the SQI only takes the ECG signal as input. This is a deliberate decision as ECG is a more sensitive and faster responding biosignal than EDA. If good quality ECG can be obtained, we can safely assume this to be the case for EDA as well. This assumption can be further supported by the fact that the majority of the noise experienced in IM-TWIN comes from large body movements and that the electrodes of both signals are placed very close to each other. As such, movement artifacts in one signal are highly correlated to artifacts in the other biosignal.*”