Improving Electrodermal Activity Quality Index by Unsupervised Pre-training

Luoluo Liu, Linda M. Eerikäinen, Octavian Cota, and Mark van Gastel

***Abstract*—Electrodermal activity (EDA) is a widely used physiological marker for assessing sympathetic nervous system activation, emotional arousal, and stress. However, EDA signals acquired from wearable devices are often degraded by motion artifacts and connectivity issues, posing significant challenges for reliable interpretation.**

In this study, we present a novel efficient and enhanced automatic EDA signal quality index system inspired by training paradigms for modern large language models. We first pre-train model with unsupervised tasks, specifically denoising and forecasting, followed by supervised fine-tuning using a lightweight backbone model and a publicly available expert-labeled dataset. Our method demonstrates an ROC-AUC of 0.851 on the EDABE benchmark dataset, an 8% improvement to previous models, while reducing the number of supervised training epochs by half, therefore improving the efficiency.

The proposed model maintains strong performance at low input sampling rates (4 Hz), compatible with commonly used wearable EDA sensors, and has a compact footprint of less than 0.3 MB (approximately 1% of a representative baseline model), enabling real-time inference on resource-constrained edge devices. Unsupervised pre-training can reduce reliance on large labeled datasets while improving performance. These characteristics support scalable, low-power EDA signal quality assessment and make the framework well suited for continuous physiological monitoring in wearable sensor systems.

***Index Terms*—Electrodermal activity (EDA), Galvanic skin response (GSR), motion artifacts, real-time prediction, signal quality index (SQI), unsupervised pre-training, wearable health solution.**

I. Introduction

Electrodermal activity (EDA), also known as Galvanic skin response (GSR), is a measure of the electrical conductivity of the skin, modulated by the activity of sweat glands [1]. Sweat gland activity responds to psychological stimuli and reflects the activity of the sympathetic nervous system [2]. Increased sweat production, in turn, enhances skin electrical conductivity, which is captured in EDA signals. Therefore, EDA has been studied for stress [3], pain [4], and sleep quality [5], [6]. For continuous monitoring applications, EDA is commonly measured on the wrist [7], [8]. For the highest accuracy, however, fingers or palms are prefered due to their highest density of eccrine sweat glands [9], [10].

Accurate EDA analysis requires distinguishing high-quality signal segments from those corrupted by artifacts. Artifact contamination is inevitable in daily-life monitoring, making automated EDA quality detection essential. Several methods have been proposed: Kleckner et al. [11] applied four rules to filtered EDA and temperature data, while others extract statistical features, Skin Conductance Responses features, and wavelet coefficients from the EDA signal to classify segments as "artifact" or "non-artifact" [12]–[16].

The most recent studies of EDA motion artifact prediction are based on deep learning approaches. Llanes-Jurado et al. [17] proposed a combined long short-term memory (LSTM) and a 1D convolutional neural networks (CNN) model. Kong et al. [18] utilized a 1D U-Net model taking both time- and frequency-domain representations as inputs.

These state-of-the-art (SOTA) methods are trained directly on labeled data, which are often expensive and time-consuming to acquire. Both approaches rely on the only publicly available EDA motion artifact benchmark dataset, EDABE [19]. Kong et al. [18] additionally used two private datasets, suggesting that further performance improvements depend on access to more labeled data.

In this paper, we propose an approach that greatly reduces the necessary amount of labeled training data by first leveraging an unsupervised pre-training stage, followed by fine-tuning based on a smaller amount of labeled EDA. The idea is inspired by demonstrated success of utilizing unsupervised pre-training in natural language models. For example, BERT employs Masked Language Modeling, in which a subset of input tokens is randomly masked and the model is pre-trained to predict the masked tokens [20]. In contrast, GPT-1 [21] and GPT-2 are pre-trained using an autoregressive forecasting task, where the model predicts the next token in a sequence [22]. In addition, unsupervised pre-training has been successfully explored in large-scale wearable foundation models [23] [24], [25]. These studies demonstrate that pre-training models with unsupervised tasks and then fine-tuning them for specific tasks such as activity classification [23] and disease prediction [24] is an effective strategy.

Unlike large wearable models with many parameters that require cloud resources and massive datasets, we show that small models trainable on a single machine for a single modality can improve state-of-the-art performance. From Table I, our method improves benchmark ROC-AUC by approximately 8% ($0.788 \rightarrow 0.851$), and reduces supervised training time by half using only one public EDA dataset. The backbone model is lightweight (68k parameters, < 0.3 MB) and has a low input sampling rate requirement(4 Hz, compared to some benchmarks requiring 128 Hz), making it suitable for real-time deployment on wearable health solutions.

Luoluo Liu is with Philips North America, Cambridge, USA;

Linda M. Eerikäinen, Octavian Cota, and Mark van Gastel are with Philips, The Netherlands;

Mark is also with the Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands.

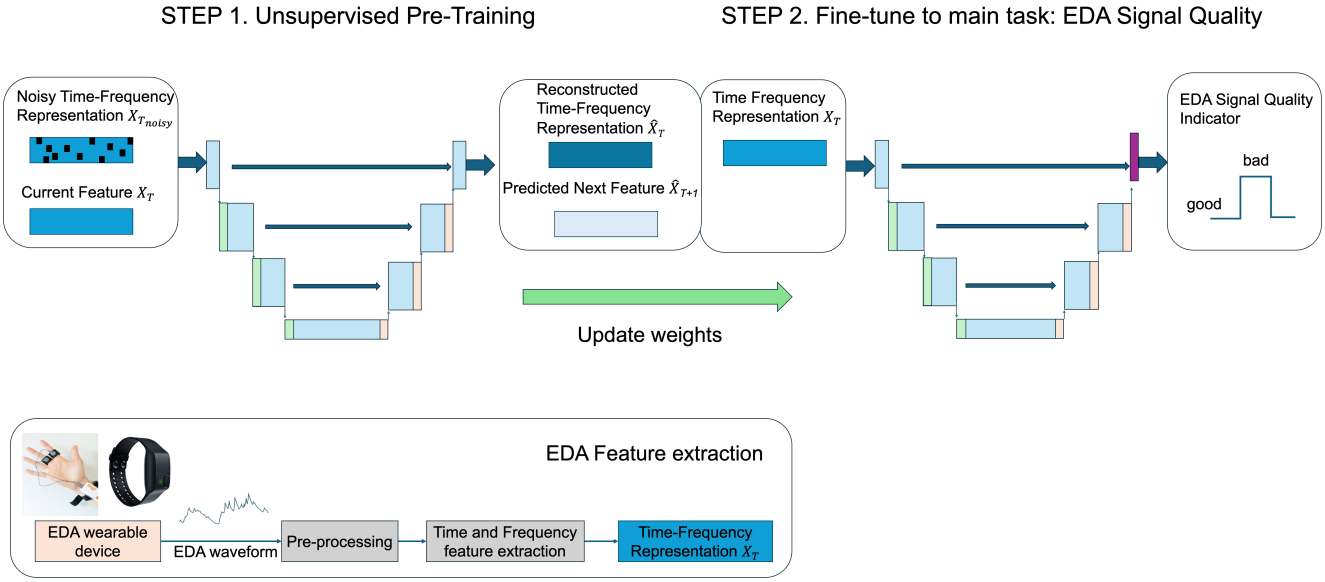


Fig. 1: Framework Illustration: The model is pre-trained with unsupervised tasks, including denoising (reconstructing 10% randomly missing values) and forecasting (predicting the next 12.5% of time-frequency features). Subsequently, supervised fine-tuning is performed using expert-labeled data to predict the EDA Signal Quality Index (SQI), indicating motion artifacts. Feature extraction from raw EDA waveforms are detailed in Methods II-C, providing inputs for the model.

TABLE I: Summary of EDA SQI methods

Method	Memory	Sampling Frequency	Train Epochs w. Labels	Mean Test AUC
LSTM -1DCNN [17]	26.74 MB	128 Hz	-	0.76
U-Net [18]	0.28 MB	4 Hz	100	0.788
Ours	0.28 MB	4 Hz	50	**0.851**

II. Methods

The framework of the EDA quality index system is illustrated in Figure 1. The EDA waveform from a wearable device serves as an input, and our framework generates a signal quality indicator (SQI) prediction as an output, indicating motion artifacts. Common off-the-shelf wearable device choices include Shimmer3 GSR+, Empatica E4, and other common sensors are described in Section II-A. We pre-process and extract time and frequency features from the raw EDA waveforms from a given wearable device, with implementation details summarized in Section II-C. Our model is first pre-trained in unsupervised manner and then fine-tuned to perform EDA SQI prediction utilizing expert labels.

Training inputs for unsupervised pre-training tasks are generated by simulation from features extracted from raw EDA signals. For the denoising task, we simulated random masks to corrupt the original input features by generating partially missing data. For the forecasting task, we used consecutive data windows as input-target pairs to train the model to predict future values. Example outputs of the simulator and the predictions from our model for denoising and forecasting tasks can be viewed in Figure 5 and Figure 6 respectively.

After pre-training, the model is fine-tuned using labeled datasets containing EDA signals and their corresponding signal quality indices, where good quality labels correspond to segments without motion artifacts, and bad quality labels for segments with motion artifacts. Once trained, the model can be deployed to predict the quality of EDA signals, determining whether the signal is useful. The threshold for classifying signal quality can be adjusted based on user preferences.

A. Common EDA Wearable Sensors Devices

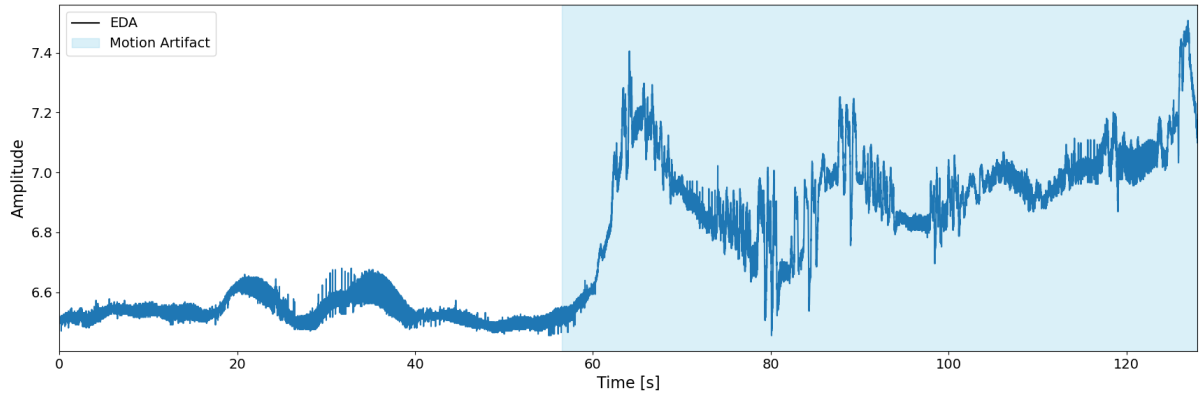
EDA is widely used to monitor sympathetic nervous system activity, stress, and emotional states, and various wearable devices have been developed to capture EDA signals and provide EDA-derived insights. Research-grade devices: Empatica E4 [26], Empatica Embrace 2 [27], and Shimmer3 GSR+ [28] provide continuous raw EDA waveforms sampled at 4, 4, and 128 Hz, respectively, supporting full wearable health solutions. For low-cost personal use, the Mindfield eSense GSR [29] provides waveforms sampled at 5 Hz for biofeedback. These devices are our top choices for EDA quality assessment. Table II summarizes these devices and their sampling frequencies, most at 5 Hz or lower, except the high-accuracy Shimmer3 GSR+ at 128 Hz.

Some consumer wearables, such as the Fitbit Sense and Garmin smartwatches, provide stress or body-response metrics derived from proprietary algorithms [30], [31]. However, these devices do not grant user access to raw EDA waveforms, and therefore are not suitable hardware options.

B. The EDABE Dataset

We utilized the "Electrodermal Activity artifact correction BEnchmark" (EDABE) [19] public dataset to train and evaluate our approach. EDABE is designed for developing

(a) Example of a raw EDA waveform with motion artifact annotation from EDABE dataset



(b) Predictions from LSTM-1D CNN, 1D U-Net and Ours utilizes unsupervised denoise pre-training task

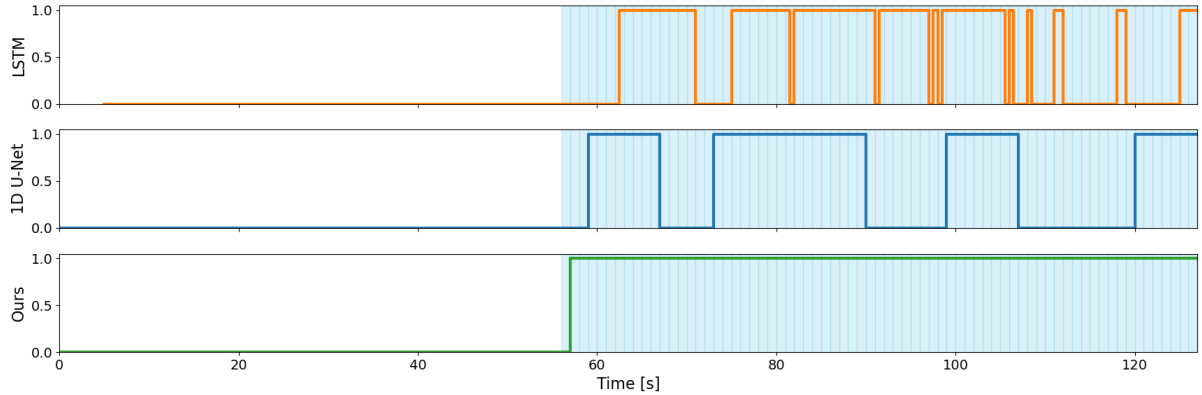


Fig. 2: **(Top)** Example of an annotated artifact in the raw EDA signal, with the blue-shaded area indicating motion artifacts as labeled by an expert. **(Bottom)** Model outputs from LSTM-1D CNN (in orange), 1D U-Net (in blue), and our method (in green). The 1D U-Net was trained from scratch for 100 epochs, while our method was fine-tuned for 50 epochs from the same 1D U-Net backbone pre-trained on the denoising task.

TABLE II: Common EDA/GSR Wearable Sensors

EDA/GSR Device	Sampling Frequency
Empatica E4 [26]	4 Hz
Empatica Embrace 2 [27]	4 Hz
Mindfield eSense GSR [29]	5 Hz
Shimmer3 GSR+ Unit [28]	Up to 128 Hz

and evaluating models for automatic artifact recognition and correction in electrodermal activity (EDA) signals. It is the first publicly available benchmark that enables systematic comparison of methods for EDA signal quality assessment and artifact detection.

The EDABE dataset consists of 74.46 hours of EDA recordings from 43 subjects, with signals affected by hand and body motion artifacts. Data were collected using a Shimmer3 GSR+ Unit at a sampling rate of 128 Hz. The data were collected during an immersive virtual reality tasks, simulating work and life situations, and a reference signal quality index indicating the presence of motion artifacts was manually annotated by two experts. The dataset contains the raw EDA signal and the manually corrected EDA signal by two experts utilizing Ledalab [32] software to perform either linear or spline interpolations on motion-affected segments.

One expert labeled EDA signals from 21 subjects while the other labeled 22 subjects. Afterwards, the whole dataset was divided randomly into a training set comprised of 33 subjects (56.27 hours) and a test set with 10 subjects (18.19 hours) by the data publisher [17]. For benchmark purposes, we adopt the same Train-Test split in our experiment, and refer them as EDABE-Train and EDABE-Test, respectively. Motion artifacts are more rare compared to non-affected areas, with approximately 10% in this dataset, and has large variance across different subjects. A detailed summary is provided in Table III.

TABLE III: Summary of EDABE [19] Dataset

Set	# Subjects	Recording	Motion Artifact (%)
Total	43	74.46 (hrs)	10.6 ± 11.6
Train	33	56.27 (hrs)	10.0 ± 11.8
Test	10	18.19 (hrs)	12.8 ± 10.6

In our work, we use raw EDA signals from the EDABE dataset as model inputs, with expert binary labels as targets.

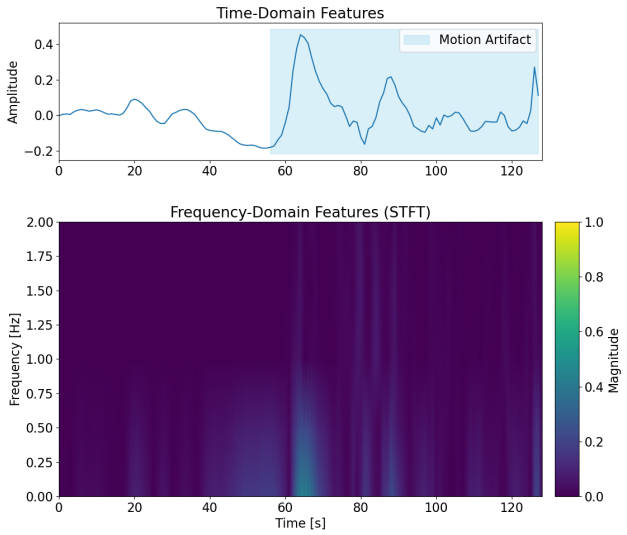


Fig. 3: Example Input Features: **(Top)** time-domain features, high-pass filtered (0.01 Hz) and downsampled from EDA raw input (Figure 2a); **(Bottom)** Frequency-domain features from short-time Fourier transform (STFT) with 2 Hz Nyquist Frequency

An example of a raw EDA signal and its corresponding expert annotations is shown in Figure 2a.

C. Time-Frequency Feature Extraction

We implemented the same pre-processing as well as feature extraction pipeline as our backbone model method, described in the U-Net EDA SQI work [18]. We first apply a high-pass filter to the raw EDA waveforms with a cutoff frequency of 0.01 Hz. Further, the signal is downsampled to 4 Hz, as evidence from the literature indicates that motion artifacts are primarily low-frequency signals [33]. This sampling frequency is widely adopted in EDA wearable devices, including popular research devices such as Empatica E4 [26].

To derive spectrograms, pre-processed signal was divided into 128-second segments. For each segment, we apply a short-time Fourier transform (STFT) with a 2-second window, and a 50% overlap (1−second). A Hann window is utilized, and the number of Fast Fourier Transform points (NFFT) equal to 8, corresponding to the Nyquist frequency at 2 Hz.

The resulting time-domain high-pass filtered signal, corresponding to the same 128-second window, was resampled and normalized. This signal was then combined with the frequency-domain features to generate a joint time-frequency representation. The combined features serve as inputs to the 1D U-Net model. An example of the time-domain and frequency-domain features is visualized in Figure 3.

D. U-Net as a Lightweight Backbone Model

U-Net is a convolutional neural network (CNN) based architecture originally developed for image segmentation tasks [34] that has many biomedical applications. In the work of Kong et al. [18], the author introduced a variant of U-Net that utilizes one-dimensional (1D) convolutional kernels to predict the signal quality index (SQI). We employ the same encoder-decoder architecture from Kong et al.'s work [18] and added a head for unsupervised pre-training.

Figure 4 illustrates the model structure of the 1D U-Net for unsupervised pre-training as well as motion artifact prediction. The input dimension of our model is 5 features combining both features from time and frequency domains, and a total of 128 timesteps extracted from one segment.

The U-Net backbone structure consists of 3 convolutional blocks, each followed by max pooling layer to reduce feature dimensions. Each convolutional block consists of two convolutional layers of kernel size 5 with a skip connection between their outputs. Each convolutional layer were followed by batch normalization and the rectifier unit (ReLU). In the encoder parts of U-Net to generates abstract feature maps of the original signals, max pooling with a size of 2 was employed to reduce the feature map size by half, while in U-Net decoder, aiming at signal reconstruction, the feature maps outputs from encoder then upsampled by a factor of 2, allowing reconstructing signal of the same input size.

In our encoder, each convolution layer contains the number of filters $8, 16, 32$, the bottleneck convolution layer contains 64 filters. In the U-Net decoder, convolution layers have $32, 16$ filters, and feature maps from the encoder convolutional blocks were directly concatenated at the same level. The final layer for unsupervised pre-trained tasks with output size 5×128, matching the input dimension. For the final layer for signal quality, it ensures an output size of 1×128 to generate one prediction per timestep (1 prediction per second).

E. Unsupervised Pre-training

We begin the unsupervised pre-training tasks by first generating data. For the denoising task, we simulate random binary masks on the original features, zeroing out 10% of the values randomly to create input-target pairs consisting of corrupted and original data segments, respectively. For the forecasting task, we construct prediction target by advancing the time window by 16 seconds relative to the input, corresponding to a 12.5% shift within a total window length of 128 seconds.

Next, we construct the pre-trained model by adapting the original 1D U-Net architecture. While the encoder retains its original structure, we modify the task-specific output layer to align with the dimensions of the input joint time-frequency features. This ensures that the outputs of the unsupervised tasks—denoising and forecasting—have the same shape as the input data. Specifically, we set the number of output channels in the final layer equal to the input feature dimension.

We use the mean squared error (MSE) loss function for both the denoising and forecasting tasks. In the denoising task, the model minimizes the difference between the reconstructed feature $\hat{\mathbf{X}}_T = f_{Unet}(\mathbf{X}_{T,noisy})$, and the original feature $\mathbf{X}_T$, whereas for forecast, the model minimizes variance between predicted signal $\hat{\mathbf{X}}_{T+1} = f_{Unet}(\mathbf{X}_T)$ to the original next

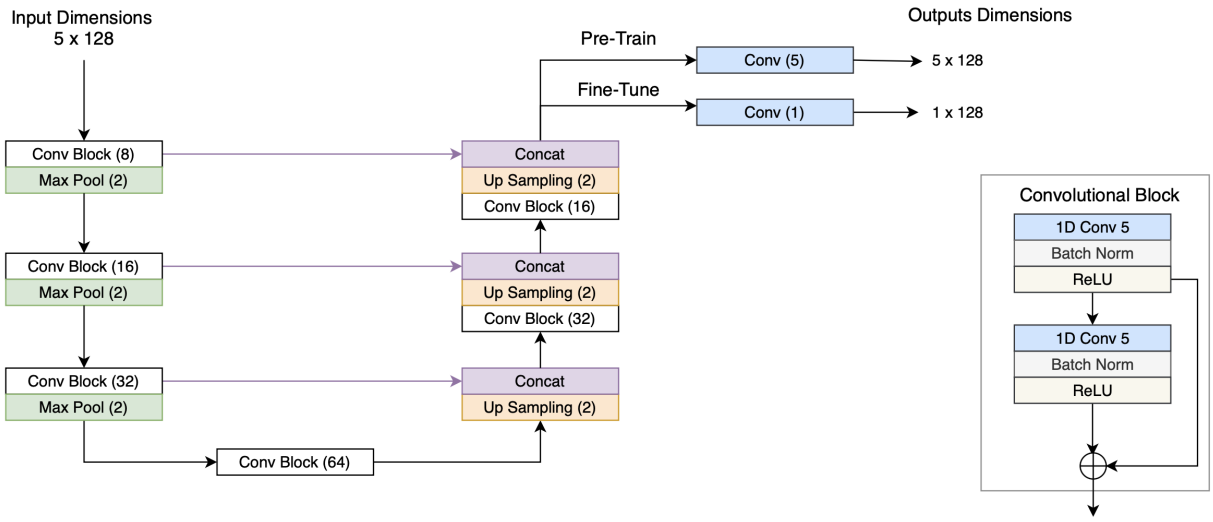


Fig. 4: U-Net model architecture for pre-training tasks (denoise, forecast) and supervised fine-tune to output Signal Quality Index. All convolution layers with kernel size are set with kernel size 5.

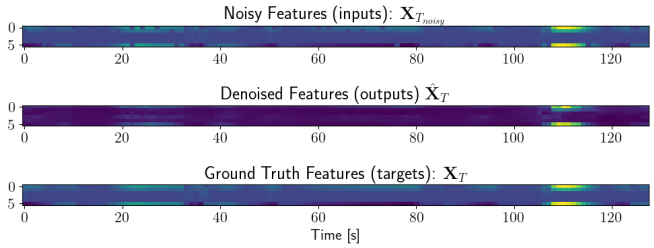


Fig. 5: Unsupervised Pre-Training, Denoise Task: The noisy example is generated by masking 10% values to zero at random locations in the input features. The model is pre-trained to reconstruct the missing values. The denoised output is from model pre-trained for 200 epochs. The mean squared error (MSE) on the entire training dataset is low, only 0.00664.

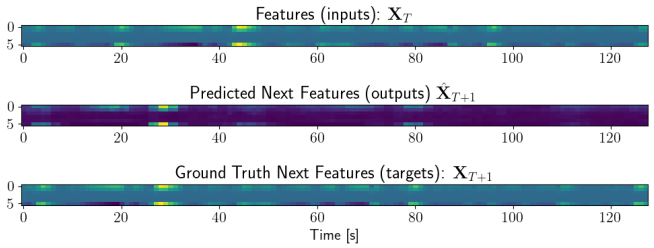


Fig. 6: Unsupervised Pre-Training, Forecast Task: The forecast target is generated by shifting the feature window 16s forward in time. The model is pre-trained to predict 12.5% of a 128-s segment. The forecasted output is generated after pre-training for 200 epochs. The MSE on the entire training dataset is low, only 0.00757.

feature $\mathbf{X}_{T+1}$. Given N total segments, the MSE loss for denoise is defined as:

$$\mathcal{L}_{\text{MSE}}^{\text{denoise}}(\mathbf{X}) = \frac{1}{N}\sum_{T=1}^{N} \|\mathbf{X}_T - f_{\text{U-Net}}(\mathbf{X}_{T,noisy})\|_2^2. \quad (1)$$

For forecast task, the same MSE loss is defined as:

$$\mathcal{L}_{\text{MSE}}^{\text{forecast}}(\mathbf{X}) = \frac{1}{N-1}\sum_{T=1}^{N-1} \|\mathbf{X}_{T+1} - f_{\text{U-Net}}(\mathbf{X}_T)\|^2. \quad (2)$$

For pre-training tasks, we use a batch size of 512 and the Adam optimizer, with a learning rate and weight decay both set to 0.0001. The denoising task is trained for 200 epochs, resulting in a low MSE of 0.0066 on the entire training dataset. Using the same training configuration for the forecasting task, the model achieves an MSE of 0.00076.

Representative results of the unsupervised tasks are illustrated in Figure 5 and Figure 6. When compared to the ground truth examples, the denoised and forecasted outputs demonstrate the model's ability to accurately reconstruct missing data and forecast future segments.

F. Fine-Tune on Motion Artifact Signal Quality Index

After pre-training the model with unsupervised tasks, we then fine-tune the model for our main task: predicting the signal quality of EDA signals. We leverage both the time-frequency features and expert-labeled signal quality annotations from the EDABE dataset.

For a direct comparison with prior work [34], the fine-tuning process follows mostly the same training configuration as the model trained from scratch, with the exception of reducing the number of training epochs to 50. We use a batch size of 512 and the Adam optimizer, with learning rate and weight decay both set to 0.0001. In addition, we adopt the same loss function as prior work [34]: an average of the Dice coefficient loss (DICE) and Binary Cross-Entropy (BCE) loss.

To derive the DICE loss, we first denote the estimated signal indicator $\hat{y}$, which is derived from input features $\mathbf{X_T}$ passing from the feedforward network U-Net $f_{\text{U-Net}}$. Then $\sigma\cdot$ denotes the activation that generates SQI: $\hat{y} = \sigma \cdot f_{\text{U-Net}}(\mathbf{X_T})$. The

DICE loss is as the following:

$$\mathcal{L}_{\text{DICE}}(y, \hat{y}) = 1 - \frac{2\sigma(\hat{y})\sigma(y) + \epsilon}{\sigma(\hat{y}) + \sigma(y) + \epsilon}, \tag{3}$$

where a small positive term ϵ set to 10^{-7} is added to improve numeric stability. The BCE loss is as follows:

$$\mathcal{L}_{\text{BCE}}(y, \hat{y}) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}). \tag{4}$$

Finally, the loss function for the EDA SQI supervised learning task takes an average over the two losses:

$$\mathcal{L}_{\text{EDA-SQI}} = 0.5\mathcal{L}_{\text{DICE}} + 0.5\mathcal{L}_{\text{BCE}}. \tag{5}$$

III. Results

We evaluate our approach using the EDABE dataset and compare its performance against the benchmark 1D U-Net implementation [18], where the network was trained from scratch, as well as another architecture LSTM-1D CNN [17]. For benchmarking, we begin with an overview of the implementations for training the backbone model U-Net from scratch. Finally, we analyze the performance differences between the LSTM-1D CNN, the original U-Net, and our model, highlighting the benefits of supervised fine-tuning from the model pre-trained with unsupervised tasks.

A. Implementation of EDA 1D U-Net benchmark

To reproduce the U-Net results for generating baseline performance for model trained from scratch, we adopted most of the parameters from the original work of Kong et al. [18]. One key difference is that we only have access to the public EDABE dataset, whereas the authors also utilized two additional private datasets. Fortunately, these private datasets are relatively small, accounting for less than 10% of the total training data. Therefore, the absence of private datasets is unlikely to significantly affect the overall training performance. Moreover, to evaluate the benefit of unsupervised pre-training, it is sufficient to train the backbone model using only EDABE. Here we simply use the same EDABE dataset for training model from scratch, pre-training and fine-tuning, for all comparison methods.

For model training, we adopted the model training parameters described by Kong et al. [18]. We utilized the Adam optimizer with a batch size of 512, a learning rate of 0.0001, and a weight decay of 0.0001. The original study reported no further improvement beyond 98 epochs; accordingly, we trained our model for 100 epochs to ensure comparable results.

B. Performance Comparison

We evaluate the cumulative performance of the original 1D U-Net method, which trains the U-Net from scratch with random initialization, and our approach utilizes unsupervised pre-training tasks. All performance metrics are reported on EDABE-Test set, from models trained from EDABE-Train set, where the train-test split is provided by the publisher and is available at the EDABE dataset website [19].

To assess cumulative performance, we calculated the ROC-AUC score for each recording in the test set. Subsequently, we computed the mean, standard deviation, median, minimum, and maximum ROC-AUC scores, as summarized in Table IV.

Our results show a significant performance improvement when utilizing unsupervised pre-training. The mean ROC-AUC score increased from 0.788 to 0.851, representing a substantial gain compared to model trained from scratch. While the standard deviation remains similar, the median ROC-AUC improved from 0.809 to 0.866. Additionally, the lowest ROC-AUC score increased from 0.652 to 0.717, and the highest ROC-AUC score rose from 0.896 to 0.974.

Beyond performance improvements, unsupervised pre-training reduces supervised training time. Training from scratch generally requires 100 epochs achieve comparable performance, as reported in prior work [18]. In contrast, fine-tuning a pre-trained model achieved improved performance with only 50 epochs, effectively halving the supervised training time. The training epochs for comparative methods are detailed in Table I.

C. Threshold-Dependent Metrics

In addition to ROC–AUC scores, we evaluate a comprehensive set of commonly used threshold-dependent metrics, as detailed in Table V. Thresholds τ are selected via a grid search ranging from 0.1 to 0.9 with a step size of 0.1. The optimal threshold is chosen based on the highest G_1 score, defined as the geometric mean of sensitivity and specificity, after which the remaining metrics are reported. This threshold selection strategy is consistent with the approach used by Kong et al. [18]. In practical deployment, users may select alternative thresholds according to application-specific criteria, such as a targeted false alarm rate. As shown in Table V, utilizing the pre-trained model yields a substantial improvement in sensitivity, increasing from 0.47 when trained from scratch to 0.78.

In binary classification tasks, there is often a trade-off between the True Positive Rate (sensitivity) and True Negative Rate (specificity), particularly in imbalanced problems where one class is much more prevalent than the other. In the case of EDA motion artifact prediction, the dataset exhibits such imbalance, with 10% positive prevalence in the training set and 12.8% in the test set.

To measure joint performance of sensitivity and specificity, G_1, balanced accuracy, and F_1 metrics are indicative, and we show the improvement of utilizing a pre-trained model for all those metrics. Fine-tuning from the model pre-trained on the denoising task yields the highest G_1 score at 0.75, compared to the one trained from scratch at 0.64. In addition, U-Net from pretrained denoise task outperforms the benchmark on Balanced Accuracy scored at 0.77 compared to without pre-training at 0.69. For F_1 score, the model pretrained with the forecasting task outperforms other methods. A similar F_1 score is achieved by the model pre-trained with a combination of forecast and denoising tasks.

This is an evidence that pre-trained models can increase performance of both predicting positive and negatives compared to the base model. Model performance metrics indicate a reduction of false positives and false negatives by utilizing

TABLE IV: Performance of ROC-AUC score on EDABE-Test recordings, * are reported performance from the cited paper, † are performance reported from our implementation. For all performance reported below, EDA data are trained from raw data from EDABE. Fine-tuned from Denoise pre-train task achieves the highest average AUC at 0.851, higher than EDABE by 12%, and Unet by 8%, with the model size only approximately 1% of EDABE model.

ROC-AUC on EDABE-Test	Mean (Std)	Median	Min	Max
LSTM-1DCNN* [17]	0.76 (0.060)	-	-	-
U-Net† [18]	0.788 (0.087)	0.809	0.652	0.896
Ours-Denoise-50	**0.851** (0.089)	**0.866**	**0.717**	**0.974**
Ours-Forecast-50	0.832 (0.088)	0.855	0.694	0.947
Ours-Denoise & Forecast-200	0.831 (0.082)	0.850	0.712	0.946

TABLE V: Record-level performance metrics on the EDABE-Test set for various algorithms. Results for single pre-training tasks — Denoise or Forecast — are reported after 50 epochs of supervised fine-tuning with expert labels. Additionally, the jointly pre-trained Denoise-and-Forecast model demonstrated reduced variance in performance but did not achieve more improvement compared to single task. Supervised fine-tuning results for the joint model are presented after 50, 100, 150, and 200 epochs, denoted as DF-50, DF-100, DF-150, and DF-200, respectively.

Method	τ	G1	BalAcc	F1	Accuracy	Sensitivity	Specificity	DICE	Kappa
LSTM-1DCNN*	0.2	-	-	-	**0.88** (0.09)	0.65 (0.16)	0.89 (0.17)	0.57 (0.07)	**0.49** (0.08)
Unet†	0.1	0.64(0.14)	0.69(0.08)	0.42(0.15)	0.87(0.05)	0.47(0.2)	**0.91**(0.06)	0.42(0.15)	0.35(0.12)
Ours-Denoise-50	0.2	**0.75**(0.10)	**0.77**(0.08)	0.43(0.13)	0.78(0.08)	**0.78**(0.2)	0.76(0.13)	0.43(0.13)	0.31(0.09)
Ours-Forecast-50	0.1	0.68(0.14)	0.72(0.09)	**0.46**(0.15)	0.87(0.05)	0.54(0.22)	0.90(0.08)	**0.46**(0.15)	0.38(0.13)
Ours (MTL):									
DF-50	0.1	0.70(0.12)	0.73(0.09)	0.41(0.14)	0.81(0.07)	0.64(0.22)	0.81(0.11)	0.41(0.14)	0.3(0.11)
DF-100	0.1	0.69(0.13)	0.72(0.09)	0.44(0.15)	0.85(0.05)	0.58(0.22)	0.87(0.08)	0.44(0.15)	0.35(0.12)
DF-150	0.1	0.69(0.13)	0.73(0.09)	0.45(0.14)	0.86(0.05)	0.58(0.21)	0.88(0.07)	0.45(0.14)	0.36(0.11)
DF-200	0.1	0.68(0.13)	0.72(0.09)	0.45(0.15)	0.87(0.04)	0.53(0.21)	0.90(0.06)	0.45(0.15)	0.38(0.12)

pre-trained tasks compared to directly training from scratch on the supervised learning task.

D. Visualizations

To better understand the benefits of unsupervised pre-training, we visualize model predictions on 128-second segments from different subjects from EDABE-Test sets comparing the outputs probabilities of the 1D U-Net trained from scratch (100 epochs) and the 1D U-Net fine-tuned from a pre-trained denoise task (50 epochs). These visualizations, shown in Figure 2b and Figure 7, highlight performance improvements achieved through unsupervised pre-training.

In Figure 2b, the reference labels show almost half signal in this segment are motion-free and half motion affected. Our fine-tuned model successfully identifies the whole region as having artifacts, whereas the 1D U-Net trained from scratch identifies some artifacts throughout the section, but it also has many gaps within which it marks as false negative.

In Figure 7, the model trained from scratch correctly predicts the first peak at 30–40 seconds but generates false positives around 45 seconds and misses peaks around 80 seconds. In contrast, the model fine-tuned from the pre-trained task accurately identifies the peak at 30 seconds, and although it produces smaller false positives after 40 seconds, these can be reduced by increasing the decision threshold. It also correctly predicts peaks around 80 and 105 seconds, which the model trained from scratch missed.

In general, utilizing unsupervised pre-training is able to reduce false positive and false negatives. Our model pre-trained with denoise task yields higher prediction probability values on average than the model trained from scratch, resulting in a relative higher optimal decision threshold.

IV. Discussion

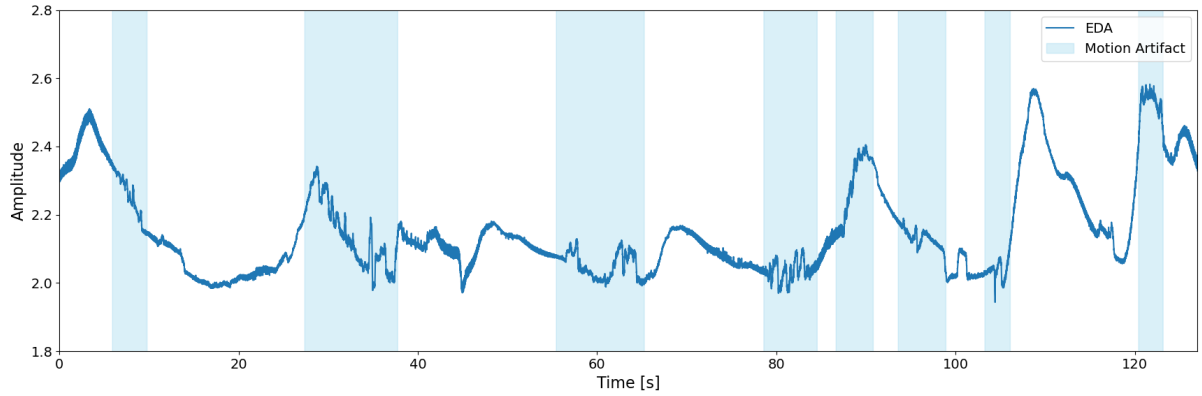
In this work, we have presented an automatic EDA signal quality index system that leverages unsupervised pre-training. In this section, we will discuss various task choices for pre-training our EDA model, and how they vary different performance metrics. In addition, we are going to discuss two important aspects in production 1) a lightweight and therefore deployment friendly model, 2) low requirement on sampling frequency, making the framework suitable to most EDA wearable devices. Both model design choices make the proposed framework desirable for real-time deployment on wrist-based EDA solutions. Finally, we touch on broader future potential of this work, with its potential on multimodal solution as well as EDA based stress predictions.

A. Pre-train tasks choices: Denoise vs Forecast

For pre-training models, multiple unsupervised tasks are commonly chosen such as denoise, forecast, self-contrastive learning. In this work, we design our unsupervised pre-training tasks to be denoising, forecasting, and a combination of both.

1) Denoise vs Forecast: From AUC performance, we could see that if we are training a single task, then model pre-trained on denoising task performs better than the one trained on forecasting task. This is logical beause, in our setup, it is easier to denoise corrupted values. Denoise task missing data ratio is 10% compared to a forecasting task where the model needs to learn to predict the features from the next batch,

(a) Raw EDA Signal with motion artifact annotation from EDABE dataset



(b) **Top**: 1D U-Net from scratch (100 epochs); **Bottom**: Ours Fine-tuned 50 epochs from a model pre-trained on denoising

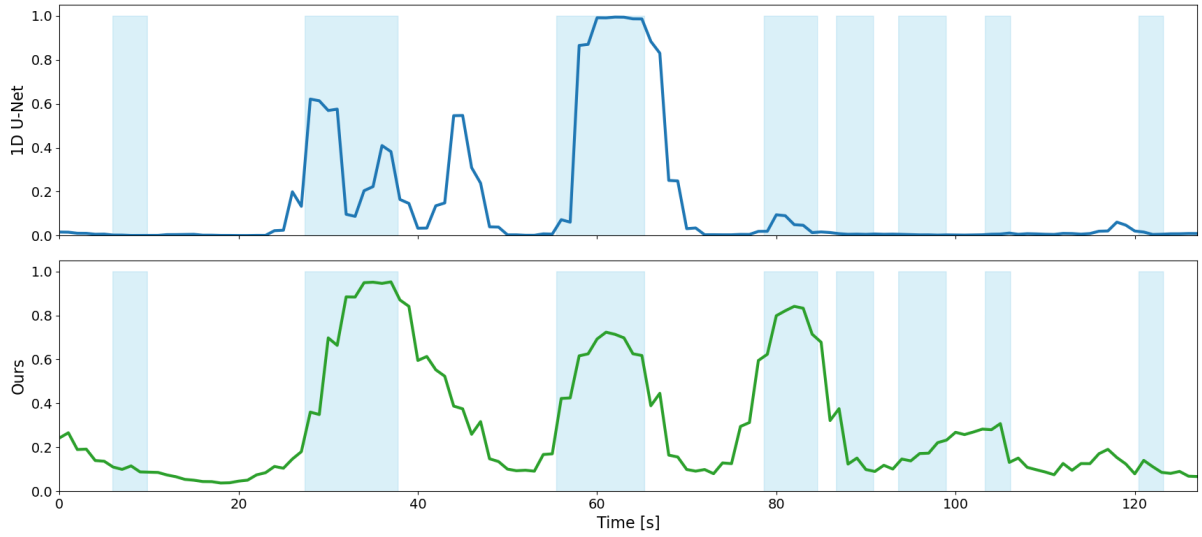


Fig. 7: Visualization comparisons of a U-Net trained from scratch (in blue) vs Ours (in green). The model trained from scratch tends to produce more false positives at 40-50s, and fails to identify positives around 80-90s. Our fine-tuned model reduces false positives and false negatives.

therefore with $1/8 = 12.5\%$ of unseen data and 87.5% of an overlap between the original features and the ground truth of the forecast features.

In addition, forecast with $1/8$ new information is generally more difficult than denoise as the later can take advantage of signal's local information. Especially when signal is not periodic as other modalities such as ECG, PPG (one EDA example shown in Figure 6), forecast is a lot harder than denoise task. In denoising task, since it is not casual within a window, neighborhood information can be used for denoise signal, it is an easier task than forecast non-periodic signals.

As shown in Table IV and Table V, pre-training on the denoising task leads to improved performance in ROC-AUC, $G1$, Balanced Accuracy, and Sensitivity compared to training from scratch. Meanwhile, pre-training on the forecasting task results in higher scores for $F1$, Accuracy, Specificity, DICE, and Kappa. From accumulative metrics such as ROC-AUC, pre-training on denoise task is more desirable.

2) Pre-Train with Both tasks: Reduced AUC Variance at the Cost of More Complex Pre-Training and Longer Fine-Tuning: Pre-training the model with both denoising and forecasting tasks results in a some reduction in the variance of AUC scores on SQI. In terms of other threshold dependent performance metrics, the performance of pre-trained both tasks are in between of the models pre-trained individually on each task.

To enable simultaneous learning of both tasks, we pre-train the model using a multi-task learning (MTL) framework. Specifically, we define a composite loss function as the sum of the mean squared error losses for the denoising and forecasting tasks. During training, we observe that the denoising loss initially exceeds the forecasting loss, but both decline consistently over the epochs, indicating effective learning for both tasks.

While MTL improves stability and reduces variance in performance, it also introduces a higher computational cost, requiring longer fine-tuning times compared to single-task models (200 vs 50 epochs). Moreover, the observed variance

reduction is modest, partly due to the limited EDABE test set size. Evaluating MTL on larger and more diverse datasets may further reveal its potential benefits and limitations.

B. Deployment-Friendly Designs: small model memory and low sampling frequency requirement

When it comes to making this algorithm feasible for real-time deployment, two important aspects are: small memory and low input EDA/GSR sampling frequency requirement.

1) Comparison of Model Sizes and Memory Consumptions: The backbone model that we use utilizes 1-D convolutions and 1-D pooling. As a result, the number of parameters is much smaller compared to similar architectures using 2D convolutions or LSTM-based framework. Consequently, it has significantly lower power consumption compared to other methods. The model requires only 0.28MB of memory as opposed to 26.74MB for LSTM approach and 54.59MB for 2D U-Net approach proposed by [17]. Small model size promotes efficient training as well as low cost for hosting service, making it preferable for potentially hosting solutions on wearable devices.

2) Potential Extension to Wrist-Based Solutions: In contrast to the LSTM-1D CNN method, which requires EDA signals sampled at 128 Hz, the proposed model operates effectively with input data sampled at only 4 Hz. This significantly lower sampling frequency aligns with the specifications of widely used wrist-worn EDA devices, such as those developed by Empatica, which typically provide EDA signals at 4 Hz. The reduced data rate requirement of the proposed model suggests strong potential for real-time deployment on wearable platforms for motion artifact prediction.

C. Future Work

Beyond performance improvements, our method offers several practical advantages. Training from a pre-trained model reduced the time required for developing EDA quality assessment models significantly. By leveraging unsupervised pre-training, our approach minimizes reliance on costly expert-labeled data. Furthermore, the pre-trained model framework enables knowledge transfer across datasets and devices, making it scalable for future wearable health applications.

Our work aligns with the recent trends in wearable AI, where companies such as Google, Apple, and Nokia Bell Labs have explored foundation models for physiological signals from wearable devices. Our advantage and special offerings are: *i)* Unlike previous approaches that primarily focus on PPG and ECG signals, our method explicitly incorporates EDA, addressing a crucial gap in wearable health technology. *ii)* our backbone model is lightweight to avoid overfitting, as EDA datasets are less available, and amount are smaller than ECG, PPG signals. *iii)* Pre-trained denoising task is suitable for non-periodic signals EDA, has shown strong performance compared to common pre-train tasks utilizes prediction.

Future work includes using the model to predict stress from EDA signals, as the secondary study goal of EDABE data collection is to measure stress from VR study [17]. The extension to stress research could be valuable if protocol or stress timestamps would be made available publically.

Additionally, future research could explore extending our pre-trained model to multi-modal learning, incorporating additional physiological signals for more comprehensive wearable health solutions. Overall, this study demonstrates the potential of pre-trained models in EDA signal analysis, contributing to more reliable and scalable physiological monitoring solutions.

V. Conclusion

In this work, we propose fine-tuning on unsupervised pre-training tasks, including denoising and forecasting, to enhance the performance of EDA signal quality assessment from a lightweight backbone model. This approach addresses a key challenge—the scarcity of expert-labeled EDA waveforms for motion artifacts, which limits training an accurate fully supervised model.

By first pre-training our model with unsupervised task, and then fine-tuning it for quality index scoring, we achieved substantial performance gains over models trained from scratch. Our evaluation on the EDABE dataset demonstrated an 8% increase in mean ROC-AUC, along with consistent gains in other performance metrics. Notably, for non-periodic signals such as EDA, denoising proved to be a more effective pre-training task than forecasting. Due to our lightweight design and low sampling frequency requirement, the model is highly suitable for real-time deployment in wearable health monitoring solutions.

References

[1] W. Boucsein, *Electrodermal Activity*, 2nd ed. Springer, 2012.

[2] H. D. Critchley, "Review: Electrodermal responses: What happens in the brain," *The Neuroscientist*, vol. 8, no. 2, pp. 132–142, 2002, pMID: 11954558. [Online]. Available: https://doi.org/10.1177/107385840200800209

[3] T. Reinhardt, C. Schmahl, S. Wüst, and M. Bohus, "Salivary cortisol, heart rate, electrodermal activity and subjective stress responses to the mannheim multicomponent stress test (mmst)," *Psychiatry Research*, vol. 198, no. 1, pp. 106–111, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165178111007931

[4] H. F. Posada-Quintero, Y. Kong, K. Nguyen, C. Tran, L. Beardslee, L. Chen, T. Guo, X. Cong, B. Feng, and K. H. Chon, "Using electrodermal activity to validate multilevel pain stimulation in healthy volunteers evoked by thermal grills," *Am J Physiol Regul Integr Comp Physiol.*, September 2020.

[5] A. Sano and R. W. Picard, "Toward a taxonomy of autonomic sleep patterns with electrodermal activity," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011, pp. 777–780.

[6] A. Sano, R. W. Picard, and R. Stickgold, "Quantitative analysis of wrist electrodermal activity during sleep," *International Journal of Psychophysiology*, vol. 94, no. 3, pp. 382–389, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167876014016237

[7] M. van Dooren, J. G.-J. de Vries, and J. H. Janssen, "Emotional sweating across the body: Comparing 16 different skin conductance measurement locations," *Physiology & Behavior*, vol. 106, no. 2, pp. 298–304, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031938412000613

[8] D. van der Mee, M. Gevonden, J. Westerink, and E. de Geus, "Validity of electrodermal activity-based measures of sympathetic nervous system activity from a wrist-worn device," *International Journal of Psychophysiology*, vol. 168, pp. 52–64, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167876021008461

[9] S. A. Shields, K. A. MacDowell, S. B. Fairchild, and M. L. Campbell, "Is mediation of sweating cholinergic, adrenergic, or both? A comment on the literature," *Psychophysiology*, vol. 24, no. 3, pp. 312–319, 1987. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.1987.tb00301.x

[10] H. F. Posada-Quintero and K. H. Chon, "Innovations in electrodermal activity data collection and signal processing: A systematic review," *Sensors*, vol. 20, no. 2, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/2/479

[11] I. R. Kleckner, R. M. Jones, O. Wilder-Smith, J. B. Wormwood, M. Akcakaya, K. S. Quigley, C. Lord, and M. S. Goodwin, "Simple, transparent, and flexible automated quality assessment procedures for ambulatory electrodermal activity data," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 7, pp. 1460–1467, 2018.

[12] S. Taylor, N. Jaques, W. Chen, S. Fedor, A. Sano, and R. Picard, "Automatic identification of artifacts in electrodermal activity data," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 1934–1937.

[13] Y. Zhang, M. Haghdan, and K. S. Xu, "Unsupervised motion artifact detection in wrist-measured electrodermal activity data," in *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, ser. ISWC '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 54–57. [Online]. Available: https://doi.org/10.1145/3123021.3123054

[14] S. Gashi, E. Di Lascio, B. Stancu, V. D. Swain, V. Mishra, M. Gjoreski, and S. Santini, "Detection of artifacts in ambulatory electrodermal activity data," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 2, Jun. 2020. [Online]. Available: https://doi.org/10.1145/3397316

[15] S. Subramanian, B. Tseng, R. Barbieri, and E. N. Brown, "An unsupervised automated paradigm for artifact removal from electrodermal activity in an uncontrolled clinical setting," *Physiological Measurement*, vol. 43, no. 11, p. 115005, nov 2022. [Online]. Available: https://dx.doi.org/10.1088/1361-6579/ac92bd

[16] M.-B. Hossain, H. F. Posada-Quintero, Y. Kong, R. McNaboe, and K. H. Chon, "Automatic motion artifact detection in electrodermal activity data using machine learning," *Biomedical Signal Processing and Control*, vol. 74, p. 103483, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809422000052

[17] J. Llanes-Jurado, L. A. Carrasco-Ribelles, M. Alcañiz, E. Soria-Olivas, and J. Marín-Morales, "Automatic artifact recognition and correction for electrodermal activity based on LSTM-CNN models," *Expert Systems with Applications*, vol. 230, p. 120581, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423010837

[18] Y. Kong, M. B. Hossain, A. Peitzsch, H. F. Posada-Quintero, and K. H. Chon, "Automatic motion artifact detection in electrodermal activity signals using 1d u-net architecture," *Computers in Biology and Medicine*, vol. 182, p. 109139, 2024. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2024.109139

[19] J. Llanes-Jurado, L. A. Carrasco-Ribelles, M. L. A. Raya, and J. Marín-Morales, "Electrodermal Activity artifact correction BEnchmark (EDABE)," 2023. [Online]. Available: https://data.mendeley.com/datasets/w8fxrg4pv5/2

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019.

[21] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[22] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019. [Online]. Available: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[23] G. Narayanswamy, X. Liu, K. Ayush, Y. Yang, X. Xu, S. Liao, J. Garrison, S. Tailor, J. Sunshine, Y. Liu, T. Althoff, S. Narayanan, P. Kohli, J. Zhan, M. Malhotra, S. Patel, S. Abdel-Ghaffar, and D. McDuff, "Scaling wearable foundation models," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2410.13638

[24] S. Abbaspourazad, O. Elachqar, A. C. Miller, S. Emrani, U. Nallasamy, and I. Shapiro, "Large-scale training of foundation models for wearable biosignals," in *Proceedings of the International Conference on Learning Representations (ICLR 2024)*, 2024.

[25] A. Pillai, D. Spathis, F. Kawsar, and M. Malekzadeh, "Papagei: Open foundation models for optical physiological signals," *arXiv*, 2024. [Online]. Available: https://arxiv.org/abs/2410.20542

[26] Empatica Inc., "Empatica E4 user manual," 2020. [Online]. Available: https://www.utwente.nl/en/bmslab/infohub/um-16-e4-usermanual-rev.2.0-20201020.pdf

[27] ——, "Embrace / embrace2 / embraceplus specifications," https://www.empatica.com/embrace/research/.

[28] Shimmer Research, "Shimmer3 GSR+ Unit." [Online]. Available: https://www.shimmersensing.com/product/shimmer3-gsr-unit/

[29] Mindfield Biosystems Ltd., "eSense skin response technical data," https://help.mindfield.de/en/skin-response-manual.

[30] Fitbit Inc., "Fitbit sense user manual." [Online]. Available: https://staticcs.fitbit.com/content/assets/help/manuals/manual_sense_en_US.pdf

[31] Garmin Ltd., "Stress monitoring in garmin smart-watches," https://www.atomicdefense.com/blogs/news/how-does-the-garmin-watch-measure-stress.

[32] M. Benedek and C. Kaernbach, "Ledalab: A Toolbox for Skin Conductance Analysis," http://www.ledalab.de, 2010.

[33] H. F. Posada-Quintero, J. P. Florian, A. D. Orjuela-Cañón, T. Aljama-Corrales, S. Charleston-Villalobos, and K. H. Chon, "Power spectral density analysis of electrodermal activity for sympathetic function assessment," *Annals of Biomedical Engineering*, vol. 44, no. 12, pp. 3124–3135, 2016. [Online]. Available: https://link.springer.com/article/10.1007/s10439-016-1606-6

[34] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, vol. 9351. Springer, 2015, pp. 234–241, available on arXiv:1505.04597 [cs.CV]. [Online]. Available: http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a