IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 21, 2020, accepted February 16, 2020, date of publication February 27, 2020, date of current version March 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2976601

Optimizing CNN Hyperparameters for Mental Fatigue Assessment in Demanding Maritime Operations

THIAGO GABRIEL MONTEIRO[1], CHARLOTTE SKOURUP[2], AND HOUXIANG ZHANG[1], (Senior Member, IEEE)

[1]Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology, 6009 Ålesund, Norway
[2]Products and Services R&D, Oil, Gas and Chemicals, ABB AS, 0666 Oslo, Norway

Corresponding author: Thiago Gabriel Monteiro (thiago.g.monteiro@ntnu.no)

This work was supported in part by the Project SFI Offshore Mechatronics funded by Norway Research Council, Norway, under Project 237896, in part by the INTPART Subsea - Subproject USP/NTNU under Grant NFR 261824, and in part by the Project SFI Marine Operations funded by Norway Research Council, Norway, under Project 237929.

ABSTRACT Human-related issues play an important role in accidents and causalities in demanding maritime operations. The industry lacks an approach capable of preventively assessing maritime operators' mental fatigue and awareness levels before accidents happen. Aiming to reduce intrusiveness, we focused on improving the mental fatigue assessment capabilities of a combination of electroencephalogram and electrocardiogram sensors by investigating the optimization of convolutional neural networks by Bayesian optimization with Gaussian process. We proposed a mapping function to optimize the network structure without the need for a tree-like structure to define the domain of variables for the optimization process. We applied the proposed approach in a simulated vessel piloting task. Even though the mental fatigue assessment for the cross-subject case is a complex classification task, the trained convolutional neural network could achieve good generalization performance (97.6% test accuracy). Finally, we also proposed a method to improve the depiction of the mental fatigue build up process. The framework presented in this work can contribute for reducing accident risk in maritime operations by improving the accuracy and assessment quality of neural network-based mental fatigue assessment tools.

INDEX TERMS Electrocardiography, electroencephalography, human factors, mental fatigue, neural networks.

I. INTRODUCTION

Humans and human-related issues are the leading causes of causalities in the maritime industry [1]–[3]. While the industry is increasingly moving towards automation, completely removing humans from the operational loop is probably impossible. Although moving human operators from vessels to onshore control centers does reduce accidents risk, it does not entirely eliminate it. Thus addressing human-related issues is of extreme importance.

Mental fatigue (MF) is a key source of human error that accumulates with time, decreasing maritime operators' capacity to react to unexpected events and understand and solve problems. Besides some regulations and recommendations [4], the maritime domain lacks objective methods to assess MF and mitigate its effects on operations. Subjective methods such as questionnaires and surveys [5], [6] have limited value as they are usually biased and do not provide real-time tools to approach the problem. In this context, more objective methods are desired.

Among objective approaches, monitoring physiological signals is considered as one of the most reliable ways of assessing MF, since changes in these signals manifest before any other external sign of MF can be captured [7]. Physiological sensors commonly used in MF assessment include electrocardiogram (ECG) [8] and electroencephalogram (EEG) [9] due to the relation between variations in the MF state levels and heart rate variability and changes in the energy spectrum of brain signals. The data gathered by each individual physiological sensor can be considered a

The associate editor coordinating the review of this manuscript and approving it for publication was Long Wang.

time-series. They can be analyzed together as a multivariate time-series, providing complementary information to a more precise MF assessment task.

It also provides relevant information about fatigue, since a person's heart rate varies significantly while in different states of tiredness

EEG is probably the most used physiological measurement of MF due to the clear relation between the power spectrum characteristics in different frequency bands and MF levels

In general, time-series are noisy, highly dimensional, and non-stationary [10], which make the MF state assessment complicated. One very popular approach to assess this kind of data in classification tasks is the use of neural networks (NN). Several state-of-the-art algorithms are variations of the traditional convolutional neural network (CNN) [11], modified to reach a high level of robustness and perform well in several kinds of applications. Higher levels of robustness increase the complexity of the model and might be unnecessary for less general implementations of CNN. We would like to investigate how to tailor a simple CNN algorithm to perform optimally in our MF state assessment problem. NN in general are very complex black box functions. Their performance is defined by a set of hyperparameters that dictates how they learn, regularize, generalize, etc. The mapping between hyperparamters and performance is in general complex, unknown, and noisy. Thus optimizing NN is not easy. The most commonly used techniques for optimizing NN include random search, grid search, and manual inspection.

Random search can lead to good results, but it is not guaranteed to find even local optima. Grid search tries exhaustively all possible combinations of hyperparameter ranges defined by the user, which can be very computationally expensive as the number of hyperparameters and size of their ranges increase. Manual inspection relies on experts' knowledge to make informed decisions about how to change the hyperparameters for the next iteration of the optimization process. This approach may leave some regions of the hyperparameters space unexplored, making it difficult to find global optima.

The current paper implements an approach capable of balancing prior knowledge about different combinations of hyperparameters and a good exploration of the hyperparameter space to improve on more commonly used approaches. Bayesian optimization (BO) is a well-established probabilistic optimization approach that is effective for finding global optima of complex and noisy functions [12]. As a new step in the optimization process, BO takes into account all previously performed optimization steps, favoring promising solutions without neglecting less promising ones via a trade-off between exploitation and exploration of the hyperparameters space. In this context, we investigate the use of BO to enhance CNN performance on MF classification using physiological sensors by means of optimizing the selection of the network's hyperparameters. We also propose an approach to conditionally optimize the CNN structure parameters without the need for a tree-structured BO algorithm.

The rest of the paper unfolds as follows. Section II briefly introduces BO. Section III presents our case study and discusses its implementation details. Section IV presents the results obtained from our case study. Section VI concludes the paper and discusses plans for further work.

II. BAYESIAN OPTIMIZATION

BO is a powerful global optimization technique to optimize complex and noisy black box functions. It is especially advantageous when the objective function to be optimized is computationally expensive to evaluate [13]. This technique draws on Bayes' theorem. Let's consider the arbitrary objective function to be optimized $f : \mathcal{X} \rightarrow IR^D$ that can be evaluated at $x_i \in \mathcal{X}$, yielding observation y_i. The accumulated observations of f can be described as $\mathcal{D}_{1:t} = \{x_{1:t}, y_{1:t}\}$. Following the Bayes' theorem we can write:

$$P(f|\mathcal{D}_{1:t}) = \frac{P(\mathcal{D}_{1:t}|f) \cdot P(f)}{P(\mathcal{D}_{1:t})} \tag{1}$$

Equation 1 states that the *posterior* probability of f given a set of observations $\mathcal{D}_{1:t}$ is conditioned to the likelihood $P(\mathcal{D}_{1:t}|f)$, to the prior $P(f)$, and to the evidence $P(\mathcal{D}_{1:t})$. The *prior* expresses our knowledge about the function prior to seeing the data and the *evidence* expresses the probability of the observations without considering the function. The posterior distribution expresses our current belief about the objective function, which can be considered as an approximation of the real objective function. This approach can be applied repeatedly in an iterative way as we amass new observations in order to improve the approximation of the real objective function.

A. SEQUENTIAL MODEL-BASED OPTIMIZATION

Sequential model-based optimization (SMBO) is a formalization of the BO approach. The SMBO algorithm iteratively approximates f with a surrogate model which is cheaper to evaluate. Then the SMBO algorithm maximizes an acquisition function over the surrogate in order to find the next best point to evaluate on f. Then the set of accumulated observations $\mathcal{D}$ is updated and the algorithm is run again. This procedure is summarized in Algorithm 1.

Algorithm 1 Sequential Model-Based Optimization

1: **procedure** BAYESIAN OPTIMIZATION
2: Random sample objective function t times
3: $\mathcal{D} \leftarrow$ *Initialize:* $\{x_{1:t}, y_{1:t}\}$
4: *loop*:
5: Fit a surrogate model to $\mathcal{D}$
6: Optimize acquisition function: $x^* \leftarrow \text{argmax}_x u(x|\mathcal{D})$
7: Evaluate objective function: $y^* = f(x^*)$
8: Add $\{x^*, f(x^*)\}$ to $\mathcal{D}$
9: **goto** *loop*.

SMBO models rely on five main components. They are:

- **A domain of hyperparameters** from where we draw combinations of hyperparameters to optimize the objective.
- **An objective function** that takes in the hyperparameters as input and outputs the score we want to maximize.
- **A surrogate model of the objective function,** which is a simpler to evaluate approximation of the objective function.
- **An acquisition function** that is optimized over the surrogate model in order to find the next point to evaluate the objective function.
- **A history of pre-evaluated points** that is used to fit a surrogate model of the objective function.

Different takes on SMBO differ mostly on how the surrogate model is generated and which kind of acquisition function is optimized on the surrogate model [14]. These two aspects of SMBO will be considered next.

B. BAYESIAN OPTIMIZATION WITH GAUSSIAN PROCESS

BO approaches rely on a prior function to express our knowledge about the function before seeing the data. Several models can be used to derivate this prior, and the Gaussian process (GP) priors are among the most commonly used. Mockus [15] showed that GP priors are well-suited for BO approaches, ensuring the conditions for convergence of the method.

A GP is a generalization of a multivariate Gaussian distribution (MGD) to function (or continuous) space. In fact, any finite subset of a GP is an MGD. A GP is completely defined by its mean function $m : \mathcal{X} \rightarrow I\!R^D$ and positive definite covariance function $K : \mathcal{X} \times \mathcal{X} \rightarrow I\!R^D$. So the GP prior can be written as:

$$f \sim \mathcal{GP}(m, K) \tag{2}$$

When we draw samples from this prior, i.e., evaluating $f(x_i)$, the function returns the mean and variances of a normal distribution of the possible value of f at x_i. A very popular choice for the mean function is $m = 0$, since it simplifies the formulation of the GP while still leading to good performance [16]. In this way, the prior distribution is solely defined by the covariance function (also known as kernel function). The correct choice of kernel function is essential for the good performance of the BO algorithm since it defines the smoothness of samples drawn from the GP.

One very common family of kernel functions for machine learning problems is the Matérn covariance function, which describes the covariance between variable base on the distance d between these variables. These functions have the form:

$$K_{matern}(d) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}d}{l} \right)^{\nu} K_{\nu} \left(\frac{\sqrt{2\nu}d}{l} \right) \tag{3}$$

where Γ is the standard gamma function and K_{ν} is a modified bezel function [17]. l is a hyperparameter that controls the width of the kernel and for anisotropic models it is usually defined by a vector of automatic relevance determination (ARD) [18].

Matérn kernels become specially interesting when $\nu = p + 0.5$, where p is a non-negative integer. In this case the function evaluation becomes very simple since it is the product of an exponential and a polynomial of order p, which is p times differentiable. The most common value used for ν on machine learning problems is $\nu = 2.5$ since it keeps a good balance between the very smooth ($\nu = 3.5$) and very rough ($\nu = 0.5$) kernels [19].

The GP can be used as the prior for our Bayesian inference. We can use this prior to make predictions about our distribution and define a posterior. Knowing the observation $\{x_{1:t}, f_{1:t}\}$, a next observation f_{t+1} can be predicted by considering that $f_{1:t}$ and f_{t+1} are jointly Gaussian. This can be written as:

$$\begin{bmatrix} f_{1:t} \\ f_{t+1} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m_{1:t} \\ m_{t+1} \end{bmatrix}, \begin{bmatrix} K_{1:t} & K \\ K^T & K_{t+1} \end{bmatrix} \right) \tag{4}$$

which, using the matrix inversion lemma [20], reduces to:

$$f_{t+1}|f_{1:t} \sim \mathcal{N}(m_{t+1} + K^T K_{1:t}^{-1}(f_{1:t} - m_{1:t}),\ K_{t+1} - K^T K_{1:t}^{-1} K) \tag{5}$$

This posterior formulation is used by the BO to guide the optimization process. The BO algorithm optimizes a metric defined by an acquisition function over the posterior to decide which point x_{t+1} to evaluate next with the objective function.

C. ACQUISITION FUNCTIONS

The correct choice of acquisition function is essential for good results when using SMBO approaches. The acquisition function determines which point the SMBO algorithm should evaluate next on the objective function, dictating the directions of the optimization. When doing so, the acquisition function deals with two opposing goals. Firstly, it is designed to incentivize the exploration of regions of the hyperparameters space that were not explored yet. Secondly, it is designed to incentivize the exploitation of regions of the hyperparameters space that have high likelihood of leading to high evaluation scores of the objective function. The most commonly used acquisition functions for BOGP are probability of improvement (POI), expected improvement (EI) [21], and upper confidence bound (UCB) [22].

1) PROBABILITY OF IMPROVEMENT

This acquisition function favors increasing the probability of improving the current best objective function evaluation. POI can be formulated as:

$$u_{POI}(x; \mathcal{D}, \theta) = P(f(x; \mathcal{D}, \theta) > f(x_{\text{best}})) \tag{6}$$

where θ is a vector with the BOGP hyperparameters.

POI tends to favor only exploitation, which can lead the optimization to local optima. This negative impact can be

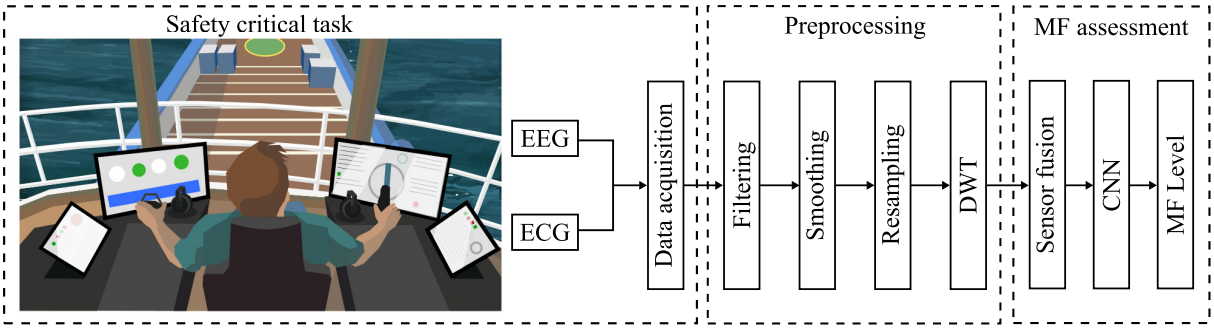

FIGURE 1. Proposed framework for mental fatigue assessment. The proposed approach aims to make the system as least invasive as possible by applying a reduced number of sensors and optimizing the CNN performance in detecting mental fatigue levels.

reduced by adding a trade-off parameter $\xi \geq 0$ to balance exploration and exploitation:

$$u_{POI}(x; \mathcal{D}, \theta) = P(f(x; \mathcal{D}, \theta) > f(x_{\text{best}}) + \xi) \quad (7)$$

2) EXPECTED IMPROVEMENT

This acquisition function rewards the objective function based on the relative magnitude of improvement. EI can be formulated as:

$$u_{EI}(x; \mathcal{D}, \theta) = E(max\{f(x; \mathcal{D}, \theta) - f(x_{\text{best}}), 0\}|\mathcal{D}) \quad (8)$$

where E represents the expectation. EI naturally provides a good balance between exploration and exploitation. However it can be further controlled by using trade-off parameters similar to the one applied on the POI function:

$$u_{EI}(x; \mathcal{D}, \theta) = E(max\{f(x; \mathcal{D}, \theta) - f(x_{\text{best}}) - \xi, 0\}|\mathcal{D}) \quad (9)$$

3) UPPER CONFIDENCE BOUND

This acquisition function explores the upper confidence bound of the surrogate model, rewarding optimism in the face of uncertainty. Consequently, the function tends to explore areas where the uncertainty is the highest. UCB can be formulated as:

$$u_{UCB}(x; \mathcal{D}, \theta) = \mu(x; \mathcal{D}, \theta) + \beta\sigma(x; \mathcal{D}, \theta) \quad (10)$$

where μ is the mean function, σ is the standard deviation and β is a trade-off parameter between exploration and exploitation. UCB explicitly defines exploitation ($\mu(x; \mathcal{D}, \theta)$) and exploration ($\sigma(x; \mathcal{D}, \theta)$) terms, which can be balanced using the trade-off parameter β.

The performance of these three acquisition functions on our CNN optimization problem is compared and discussed in Section IV.

III. MATERIALS AND METHODS

In order to investigate the performance of a BO algorithm in optimizing CNN for MF assessment in the demanding maritime operations we performed a simulated navigation experiment. In this experiment we apply a mixed method approach, where questionnaires and scenario-based experiments are used simultaneously. Fig. 1 shows the framework used to assess the fatigue state of human operators during demanding maritime operations. It presents three main phases: data acquisition, data preprocessing, and MF assessment.

This section describes our experimental setup and the most relevant points regarding data acquisition, preprocessing, and labeling. We also define a CNN as our MF assessment tool and discuss the implementation of BO for CNN structure selection.

A. CASE STUDY AND EXPERIMENTAL SETUP

The simulated navigation experiment was performed in collaboration with the Numerical Offshore Tank (TPN-USP) in São Paulo, Brazil, via the INTPART Subsea project. The experiment was conducted on a general purpose ship bridge simulator. The task consisted in navigating a large container vessel to an unloading berth on the Port of Niteroi, Brazil. The task took around 80 minutes to be completed and required moderated levels of attention from the participants due to local vessel traffic and weather conditions. At the end of the task, a complex mooring maneuver was necessary to place the vessel in the correct berth.

Six participants performed a simulation run during the morning period. All participants were males, aged 19 to 48. All were trained personnel from the Brazilian Navy, which ensures that their decisions and behavior during the experiment followed standard navigation procedures. In order to reduce the impact of external factors on the participants' MF state, we asked them to try to get 8 hours of sleep the night before the experiment and avoid the consumption of stimulants (including caffeine) or any kind of drug that could affect cognitive or motor capacities 8 hours prior to the beginning of the experiment.

During the experiment we used a set of sensors to collect physiological data from the participants. The data is collected from disparate sensors and is centralized by a micro-controller. In this case study we recorded six EEG channels and one ECG channel using the 14-channel EEG headset Emotiv Epoc+ [23] and the Electrocardiogram Sensor PRO for MySignals (eHealth Medical Development Platform) [24]. Our experiment followed the principles and guidelines of the Declaration of Helsinki and participants'

data was handled following the recomendations of the Norwegian Centre for Research Data [25].

Besides the scenario-based experiment the Karolinska Sleepiness Scale (KSS) questionnaire was used as a self-assessment tool for sleepiness estimation. Although sleepiness and MF are different concepts, they are correlated since MF is a precursor of sleepiness. Each participant indicated his self-assessed sleepiness state twice, first immediately before the beginning of the experiment and later immediately after the end of the experiment. This information was later used during the data labeling process.

B. DATA PREPROCESSING

The preprocessing stage includes filtering, smoothing, resampling, and discrete wavelet transform (DWT). During the filtering phase, artifact removal algorithms can be applied to remove unwanted perturbation from the signal. This step is not essential when using CNN, since the network provides very robust features. Any noise contaminating the data, such as power line noise, can be removed during the smoothing phase. If necessary, the channels from different sensors can be resampled to a desired frequency and aligned to ensure temporal correlation between different signals.

The ECG and EEG channels were sampled at 128 Hz. The EEG channels were decomposed using DWT to obtain the main frequency bands of clinical interest for MF assessment: delta (δ), theta (θ), alpha (α), and beta (β) [26]. Working with wavelets is advantageous in this case since this approach allows a frequency domain analysis while conserving the temporal characteristics of the data.

C. SENSOR FUSION

Although we have one ECG and six EEG channels available, there is no need to use all seven channels for the MF classification task. From the EEG channels, we select the beta sub band of electrodes AF4, F4, and O2 due to their high relevance for MF assessment [26]. We do not apply artifact removal techniques on the EEG data since CNN can extract robust features from the input data that mitigate the need for this kind of specific treatment to the EEG signals. The ECG signal acts as a complementary information source to the EEG channels, since it carries correlated MF data but in a completely uncorrelated form.

The selected channels were fused using low-level (or raw data) fusion. All data channels were aligned and the input signals were generated using a sliding window 6 seconds long with 2 seconds of overlap. The corresponding inputs from each channel were concatenated as one-dimensional input vectors (Fig. 2). From all available data in each experiment, we only used the first and last fifths of the data from each channel for training the CNN. This was due to the availability of labels for the input vectors. The labels were assigned as ''non-fatigue'' and ''fatigue'', based on the KSS questionnaire answers participants gave at the beginning and end of the experiment. The input-label pairs are fed to the CNN as the input and output data during the network training, validation, and testing.

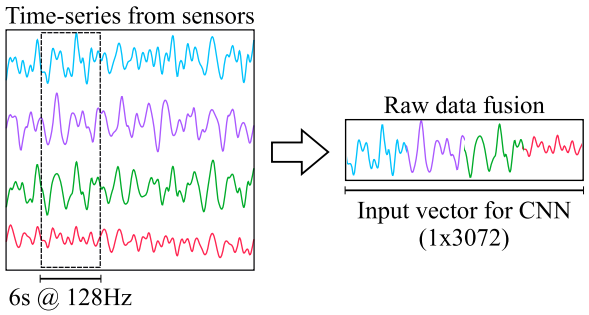

FIGURE 2. Raw data fusion scheme. Data from the different channels is aligned and segmented using a sliding window of length 6 s, with overlap of 2 s between consecutive segments. The obtained segments are then concatenated as an one-dimensional input for the CNN.

D. MENTAL FATIGUE STATE CLASSIFICATION

The MF state classification will be performed by a CNN. The CNN was initially designed to handle image classification, which requires dealing with large input data and identifying very complex features. The applications of CNN have evolved through the years and nowadays it is commonly used for time-series classification [11]. The general structure of a CNN includes an input layer, one or more feature extraction blocks, and a classification (or output) layer. Each feature extraction block is generally composed of a convolutional layer, followed by an activation function and a pooling layer. The convolutional layer applies filters to the input data, and is responsible for the features extraction. The activation function makes the extracted features non-linear, which is very important to ensure that the network can learn complex features. The pooling layer is responsible for dimensional reduction, reducing the amount of parameters and computations in the network.

TABLE 1. Hyperparameters ranges for CNN optimization.

parameters	range
# of layers	[1, 2, 3, 4, 5]
# of filters on layer n	[32, 64, 128, 256]
kernels size on layer n	[3, 5, 7, 9]
dropout rate	[0:0.5]
batch normalization [a]	[No, Yes]

[a] Only used for cross subject classification.

In our optimization process we are going to work with the following hyperparameters: number of layers, number of filters per layer, kernels sizes in each layer, dropout rate, and batch normalization (Table 1). The ranges tested for each hyperparameter during the optimization procedure are based on our previous experiences manually tuning CNNs for this kind of application.

E. BAYESIAN OPTIMIZATION FOR NEURAL NETWORK STRUCTURE SELECTION

The implementation of a BO approach to optimize CNN parameters is not straightforward when one considers the optimization of hyperparameters defining the CNN structure, i.e. number of layers, number of filters in each layer, and filter size. First, the acquisition function maps the bounded set to real and positive numbers, while the structural hyperparameters are always integer values. Second, the choices with respect to number of layers, number of filters in each layer, and filter sizes need to be coherent, since one CNN with two layers, number of filters $= \{32\}$ and kernel sizes $= \{9, 7, 5\}$ is not well defined. The first problem can be easily solved by using a floor-type function, which maps real and positive numbers to integer and positive values. The second problem requires a more elaborate treatment.

A common approach for optimizing NN structures is to define a tree structure in the optimization variable space that accounts for all possible network configurations. The drawback of this approach is that it increases the dimensionality of the optimization problem. Alternatively, we propose the use of mapping functions to map three variables to the full network structure. Consider optimizing a CNN with the maximum number of layers given by l_{max}. In light of the previously presented four possible options for the number of filters per layer and four possible options for the kernel sizes in each layer, there is a total of $4^{l_{max}}$ possible combinations of number of filters per layers and $4^{l_{max}}$ possible combinations of kernel sizes. The possible combination of number of filters per layer can be mapped to the integer interval $[1 : 4^{l_{max}}]$. From this mapping, the number of filters in the j^{th} layer of the *ith* combination can be recovered by:

$$nf_{ij} = 2^{\{\lfloor (j-1)/4^{(l_{max}-i)} \rfloor) - 4 \cdot \lfloor (j-1)/4^{(l_{max}+1-i)} \rfloor) + 5\}} \quad (11)$$

where $\lfloor a/b \rfloor$ denotes the integer division of a for b.

A similar equation can be derived for the kernel sizes. The possible combination of kernel sizes can be mapped to the integer interval $[1 : 4^{l_{max}}]$. From this mapping, the kernel size in the j^{th} layer of the *ith* combination can be recovered by:

$$ks_{ij} = 3 + 2 \cdot (\lfloor (j-1)/4^{(l_{max}-i)} \rfloor - 4 \cdot \lfloor (j-1)/4^{(l_{max}+1-i)} \rfloor) \quad (12)$$

These mapping functions always return a l_{max}-sized vector. The last elements of these vectors are trimmed off as necessary, according to the number of layers of the current network being evaluated.

As an example, from Equations 11 and 12, for a four-layer CNN with $l_{map} = 758$ and $k_{map} = 500$, the network structure is defined by:

$$\begin{aligned} nf_{758,1:5} &= \{128, 256, 256, 64, 64\} \\ ks_{500,1:5} &= \{5, 9, 9, 3, 9\} \\ CNN(4, 758, 500) &= \{128(5), 256(9), 256(9), 64(3)\} \end{aligned}$$

IV. RESULTS AND DISCUSSION

Physiological signals can present different patterns when comparing data from different individuals. This is especially true for signals of low amplitude and susceptible to noise such as EEG data. In order to take this factor into consideration in our analysis, we are going to perform two kinds of study: single-subject and cross-subject analysis.

A. SINGLE SUBJECT ANALYSIS

For the single-subject analysis we are going to use the CNN structure presented on Fig. 3. The network training followed a nested 20-fold cross validation approach. The input vectors for each subject were shuffled and divided into five groups. In each fold of the cross validation, three groups were selected as a training set, one group as a validation set, and one group as a test set. In this way, 20 folds are needed so all possible combinations of training, validation, and test sets are used for training the network. The results for all folds are averaged out in order to obtain the final validation and test accuracies for the network. This approach reduces the bias of a favorable selection of training, validation, and test sets and ensure a more fair analysis.

Back propagation was used to train the CNN and the training algorithm was carried out for at least 15 epochs. The validation accuracy was used for adjusting the dynamic learning rate and as a termination criteria for the training. If after five epochs of training no improvement on the validation accuracy was obtained, the learning rate would be reduced by 20%. After 15 epochs if no improvement on the validation accuracy was obtained, the training process would be terminated. After termination, the set of weights that performed the best on the validation set would be reloaded on the CNN model in order to evaluate the network test accuracy using a test set the network has never seen before. This ensures a fair assessment of its classification and generalization capabilities.

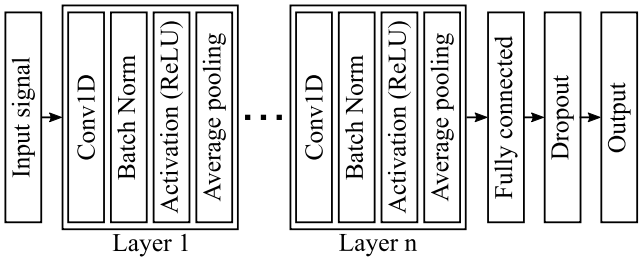

FIGURE 3. CNN structure for single-subject classification (deactivated Batch Normalization blocks) and cross-subject classification (activated Batch Normalization blocks).

For the BO, the average validation accuracy was selected as the optimization metric. The optimization variables were, first, random sampled for 40 epochs and then the BOGP was run for 400 epochs. In order to choose among the different acquisition functions presented on Section II, we performed a comparison on the performance of these three acquisition functions on optimizing the CNN for the MF state classification of Subject 1. We set up the three acquisition functions to perform with a good equilibrium between exploration and exploitation. All the cases sampled during the three optimization procedures were ranked from best to worst validation

TABLE 2. Optimization results (single subject case).

	BOGP				Random search			
	Val acc (max)	Test acc (max)	Val acc top 20 (avg ± std)	Test acc top 20 (avg ± std)	Val acc (max)	Test acc (max)	Val acc top 20 (avg ± std)	Test acc top 20 (avg ± std)
Subject 1	**0.966**	**0.951**	**0.963 ± 0.001**	0.947 ± 0.002	0.964	0.950	0.962 ± 0.001	**0.948 ± 0.001**
Subject 2	**0.967**	**0.948**	**0.965 ± 0.001**	**0.942 ± 0.002**	0.960	0.944	0.957 ± 0.001	0.937 ± 0.002
Subject 3	**0.965**	**0.947**	**0.961 ± 0.002**	**0.941 ± 0.002**	0.957	0.939	0.951 ± 0.003	0.930 ± 0.003
Subject 4	**0.951**	0.928	**0.947 ± 0.001**	**0.923 ± 0.002**	0.949	**0.931**	0.941 ± 0.004	0.916 ± 0.005
Subject 5	**0.954**	**0.935**	**0.949 ± 0.002**	**0.929 ± 0.003**	0.946	0.924	0.939 ± 0.002	0.919 ± 0.002
Subject 6	**0.945**	**0.922**	**0.937 ± 0.003**	**0.912 ± 0.005**	0.918	0.891	0.906 ± 0.006	0.877 ± 0.007

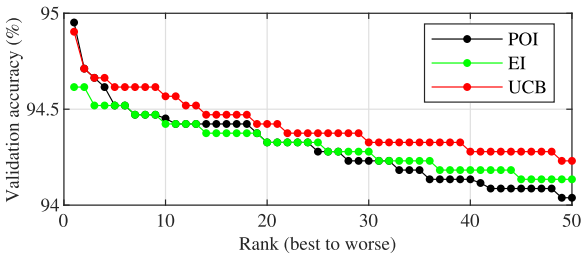

FIGURE 4. Performance of different acquisition functions on Optimizing the CNN for MF assessment of Subject 1. The plot shows the top 50 CNN results for each acquisition function, ranked by validation accuracy.

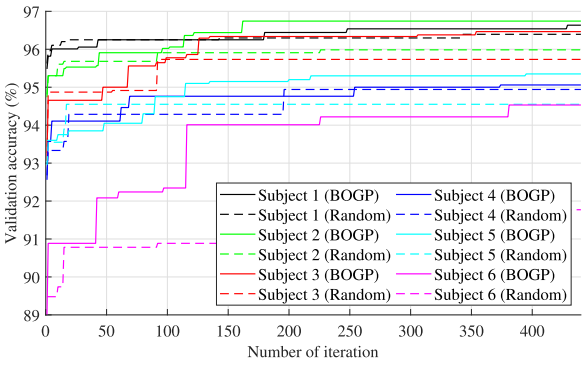

FIGURE 5. CNN training progression for single subject cases, comparing BOGP and random search for hyperparameters selection.

accuracy (Fig. 4). As we can see, although POI achieves the best validation accuracy, its overall performance is less stable than those of UCB or EI. We decided to stick with UCB as the acquisition function of choice throughout the rest of the analysis due to its overall better performance when compared to EI.

The performance of the optimizer on the CNN for classifying MF for each test subject is shown in Table 2 and Fig. 5. The BOGP optimization procedure was compared with random search for hyperparameter selection. With the same computational budget, random search is superior to other uninformed optimizers, such as grid search, and can also surpass manual search, especially in high dimensional problems. This way random search provides a natural baseline to compare against other optimization algorithms [27].

In Table 2 we present the maximum validation and test accuracies for both BOGP and random search optimization for each test subject. Since the CNN response surface is extremely noisy we also present the average of the top 20 validation and test accuracies. The best results between the correspondent BOGP and random search cases are highlighted in bold. We can see that BOGP leads to network configurations that achieved best validation accuracies for all test subjects. Random search only presented better results for the maximum test accuracy of test subject 4 and for average test accuracy for test subject 1. Here it is important to remember that the test accuracy wasn't the optimization metric, as it is only a consequence of evaluating the CNN with the chosen hyperparameters.

Fig. 5 shows how the validation accuracy for each test subject progressed during the optimization process for both BOGP and random search. For clarity we only present the boundary of the scatterplot for each test subject. It is noticeable that random search achieves best validation accuracies during the initial stages of training, but BOGP surpasses it in all cases after the algorithm gathers more information to guide the optimization process.

B. CROSS-SUBJECT ANALYSIS

In cross-subject analysis, the differences in physiological signals for different individuals need to be taken into consideration. Since this is a more complex classification task than single-subject analysis, we are going to experiment with batch normalization as an extra regularization technique.

We will also evaluate two different cases for the cross-subject analysis. Case 1 uses the same network structure as the single-subject analysis (Fig. 3). Case 2 includes the possibility for batch normalization, but lets the optimizer choose whether or not a batch normalization block is active by means of an extra optimization parameter (Table 1).

The network training followed a five-fold, cross-validation approach. The data from one subject is kept out of training to be used as a test set on the trained CNN. The data from the other five subjects is used for the cross-validation procedure, where in each fold one of the subjects is used as a validation set and the others are used as a training set. The training algorithm is based on back propagation and follows the same criteria used for the single-subject case.

The results for all folds are averaged out in order to obtain the final validation and test accuracies for the network. This approach reduces the bias of a favorable selection of training, validation, and test sets and produces a more fair analysis. For the BOGP, the average validation accuracy was selected as the optimization metric. The optimization variables were first random sampled for 40 epochs and then the BOGP was run for 360 epochs using UCB as the acquisition function.

TABLE 3. Optimization results (cross subject case).

	Val acc (max)	Test acc (max)	Val acc top 20 (avg $\pm$ std)	Test acc top 20 (avg $\pm$ std)
Case 1 - Random	0.926	0.936	0.893 ± 0.015	0.919 ± 0.009
Case 1 - BOGP	**0.949**	0.955	$\mathbf{0.921 \pm 0.013}$	0.928 ± 0.015
Case 2 - Random	0.936	0.970	0.917 ± 0.010	0.944 ± 0.012
Case 2 - BOGP	0.937	**0.976**	0.919 ± 0.010	$\mathbf{0.951 \pm 0.012}$

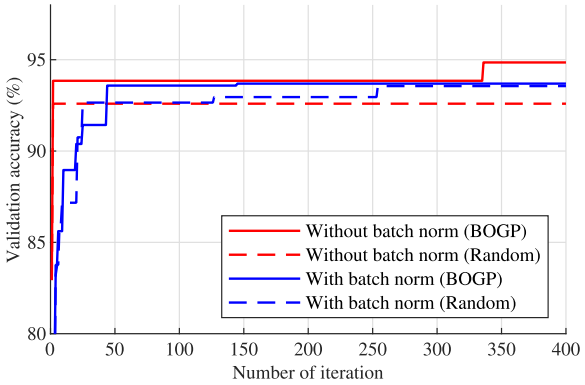

FIGURE 6. Validation accuracy progression for CNN training for cross-subject cases for hyperparameters selection. The plot compares BOGP and random search with and without batch normalization.

The optimization results for both cases are shown in Table 3 and Fig. 6 and 7. From Table 3 we can see that BOGP presented the best performances for validation and test accuracies. For Case 1 (no possibility of batch normalization) BOGP reached the best individual and average validation accuracies. For Case 2 (the optimizer chooses whether to use batch normalization) BOGP reached the best individual and average test accuracies. Fig. 6 show how the use of batch normalization impacts the validation accuracy by reducing the network overfitting while Fig. 7 shows how the regularization power of batch normalization can help the network to achieve a more general configuration that is able to perform better on new data. In both BOGP and random search, the batch normalization provided a significant increase in performance when compared with the cases without batch normalization.

C. OPTIMIZATION VARIABLES BEHAVIOR

BOGP can produce good MF classification for both single- and cross-subject cases. Although CNN are very noisy

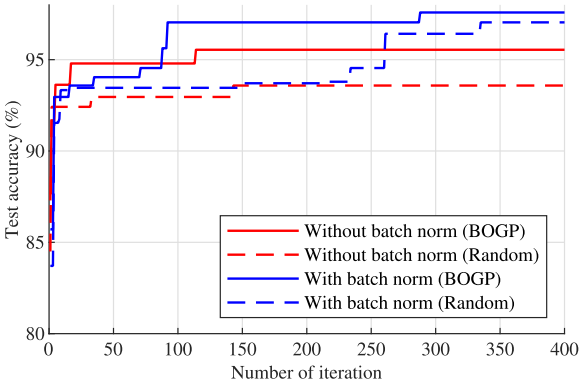

FIGURE 7. Test accuracy progression for CNN training for cross-subject cases for hyperparameters selection. The plot compares BOGP and random search with and without batch normalization.

functions, we can gain valuable insights about their behavior by analyzing how the optimization variables are related to each other during the optimization process. Since the second case used to analyze the cross-subject classification is the more complex and general case study, we will analyze its hyperparameters optimization in this section.

In Case 2, the optimization variable space is defined on $\mathcal{X} \rightarrow I\!R^6$. The relation between variables on such high dimensional space is hard to visualize. In Fig. 8 we present a parallel coordinate plot showing the relation between the five optimization variables (dropout rate, batch normalization, number of layers, number of filters per layer, and kernel sizes per layer), the optimization metric (validation accuracy) and the evaluation of the optimized CNN on new data (test accuracy). The darker blue lines represent configuration with the highest validation accuracy. On this plot we excluded the 40 random samples used to initialize the optimization process.

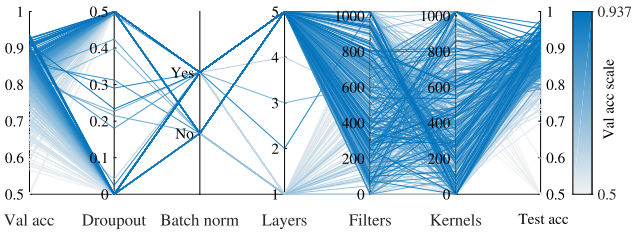

FIGURE 8. Parallel coordinate plot showing the relation between optimization variables, optimization metric (validation accuracy) and test metric (test accuracy). Darker hues represent cases with higher validation accuracy.

When analyzing Fig. 8 we can see that there is a clear relation between the validation and test accuracies. Also, the CNN structures with five layers are clearly the best. The dropout rate is in general chosen as 0 (no dropout) or 0.5 (maximum dropout rate available), with intermediate values being neglected. There is a relation between dropout rate and batch normalization variables, which is hard to see since batch normalization is a discrete variable and, consequently,

different configurations overlap each other on the plot. Additionally, for this variable space it is hard to comprehend the behavior of the filter and kernel variables, since their real meanings are hidden by the mapping function presented in Section III. We can try to clarify these points by analyzing the correlation matrix between the CNN hyperparameters.

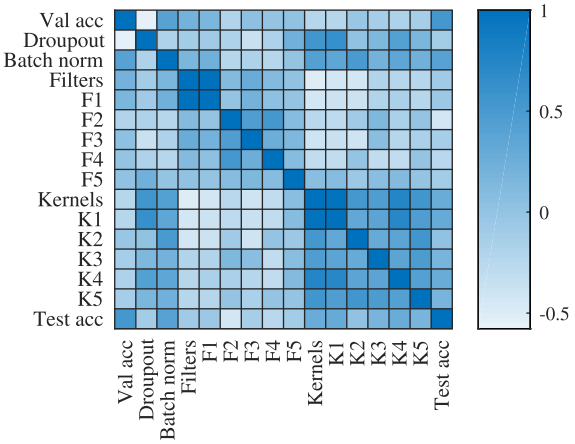

FIGURE 9. Correlation matrix for CNN hyperparameters and validation and test accuracies. The matrix includes data for the CNNs with the top 20 highest validation accuracy. Darker hues represent bigger positive correlation, while lighter hues represent bigger negative correlation.

Fig. 9 presents the correlation matrix for the 20 best-ranked CNN configurations (based on the validation accuracy). Restricting the analysis to only the best models is important when evaluating the correlation matrix, since non-optimal configurations would contaminate the correlation analysis with irrelevant data. As we can see, there is a negative correlation between the dropout rate and batch normalization, which indicates that these two variables in general are not active at the same time for a model. When analyzing the correlation between validation accuracy and dropout rate, we note an even stronger negative correlation, showing that top validation accuracies are achieved with no dropout.

Regarding the filters distribution per layer, we can see that better validation accuracies configurations tend to present more filters on the first layers, with the number of filters slightly decreasing with the depth of the layers. For kernel sizes, we find the opposite pattern, with smaller kernels on the first layer and bigger kernels on deep layers. A similar opposition is evident when comparing the correlation between filters and kernels and also by checking the crossing lines between filter and kernel variables on Fig. 8.

V. ASSESSING MENTAL FATIGUE LEVEL

After the optimization process, a good network structure can be select for the CNN responsible for the MF assessment process. After trained, the CNN can be used in real-time to assess the MF level of operator during maritime operation. As explained in Section III-C, the first fifth of the time-series from each sensor was labeled using the KSS score assigned by the participants in the beginning of each experiment and the last fifth of the time-series from each sensor was label using the KSS score assigned by the participants in the end of the experiment. This means that we do not have information about transition states between the non-fatigue and fatigue states. When using the trained CNN to assess the MF state of one operator during the whole experiment period, we obtain the result presented in Fig. 10-a.

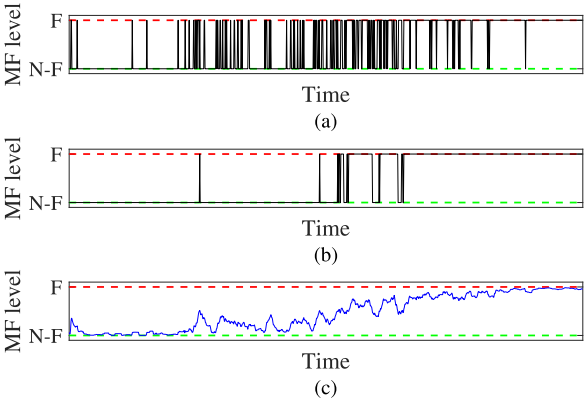

FIGURE 10. Several steps on the proposed MF level assessment approach. (a) Performance of MF assessment using the trained CNN output in new data. (b) Performance of MF assessment using the trained CNN output and a 15 step averaging window in new data. (c) Performance of MF assessment using the trained CNN probabilistic output and a 15 step averaging window in new data.

This result is unsatisfactory in two ways. First, the MF assessment is noisy due to the natural dynamic of the ECG and EEG signals and noise levels presented in the sensors data. Second, the CNN was capable of assessing the initial and final MF states very well, but there is no gradual transition between these two states. This is not the result we expect to see, since MF is a cumulative process and, therefor, should build up with time.

In order to address the first issue, a simple solution can be applied. Since we are interested in the general trend of the MF state rather than the exactly value in each time step, we can apply an average moving window that considers the current and previous N time steps. This averaging window is applied after the output from the CNN. The result of such approach considering a 15 time steps window is presented in Fig. 10-b.

For addressing the second issue we also propose a simple solution. But before approaching the proposed solution, we need to first understand how the CNN performs the classification task. The CNN output is the result of the Softmax activation function. This function maps logits input values to probabilities of that input belonging to each one of the classes in the classification problem (in this case the two MF levels). It basically represents how certain the CNN is that the input data from the physiological sensors represents each one of the MF states. The final class assigned by the CNN is the one with the highest probability.

Our proposal is to, instead of using the final class assignment to assess the MF state, use the probability distribution from the Softmax function. Is this case, with only two classes,

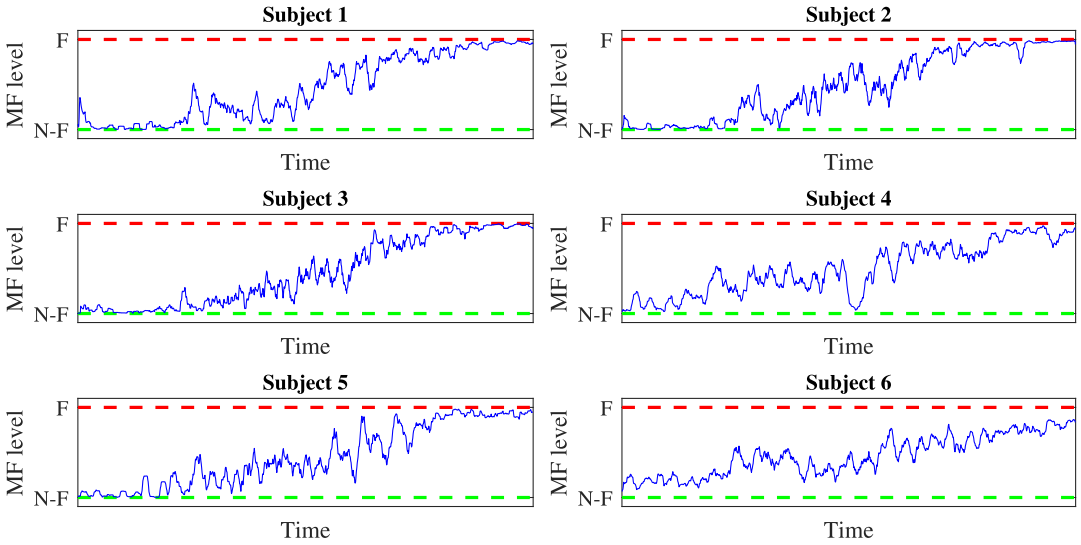

FIGURE 11. Mental fatigue level assessment for all test subjects using CNN probabilistic output and 15 time steps averaging window.

the probabilities of each class are complementary, so we just need to focus on the probability of one of the classes. Fig. 10-c present the MF assessment using the probability that the input data represents the ''fatigue'' class and a 15 time steps averaging window. A smoother transition between the two MF states is captured using this approach.

Fig. 11 shows the MF assessment for each one of the 6 Subjects in our experiments. All the cases, except for Subject 6, present a very well defined distinction between the ''non-fatigue'' and ''fatigue'' states. As we can see, the CNN has difficulties distinguishing between the two MF state for Subject 6. This may be an indication that the progression of the MF during the experiment was not very accentuated for this specific case. Besides that, the MF progression for Subject 6 is still captured in the analysis.

VI. CONCLUSION AND FUTURE WORK

Although monitoring the development of MF in maritime operators is important in order to reduce human-related accidents and causalities, the use of physiological sensors can be intrusive and hinder effective operation of equipment and systems. Thus, reducing the number of sensors and optimizing the use of data collected by them is very important.

In this work we investigate the optimization of CNN for assessing MF on vessel pilots using only EEG and ECG sensors. The optimization procedure was conducted using BOGP due to its good performance in optimizing black box functions. We also propose the use of mapping functions to provide optimization of the CNN structure while reducing the dimensionality of the optimization problem. The optimization process achieved good results on both single- and cross-subject analysis; as the more complex case, the latter is the one that really matters. On the cross-subject case the obtained average test accuracy was 95.1%.

The MF assessment using the optimized CNN did not provide a satisfactory results, since the classification was noisy and did not account for the intermediate conditions between the ''fatigue'' and ''non-fatigue'' states. In order to assess the MF level in a more gradual manner, we propose an approach based on the probabilistic output of the CNN combined with an averaging sliding window. This method showed consistent results across all test subjects and was capable of providing a MF assessment that accounts for the continuous progression of the MF state with time.

In a future work we plan to better explore the BOGP hyperparameters selection and expand the boundaries of the CNN hyperparameters space used for the optimization, since the optimizer kept variables like dropout rate and number of layers to their superior limits, which suggests that there is space for further exploration of these variables. A better discretization of the MF scale is also desirable. This can be achieved by making the participants answer the KSS questionnaire several time during the duration of the task, besides only getting the answers for the beginning and end of the experiment. Finally, we also would like to extend our case study and include more test subjects in order to ensure the obtained results are statistically significant and the proposed approach can be applied as a general tool to assess mental fatigue.

REFERENCES

[1] C. Hetherington, R. Flin, and K. Mearns, ''Safety in shipping: The human element,'' *J. Saf. Res.*, vol. 37, no. 4, pp. 401–411, Jan. 2006.

[2] J. U. Schröder-Hinrichs, ''Human and organizational factors in the maritime world—Are we keeping up to speed?'' *WMU J. Maritime Affairs*, vol. 9, pp. 1–3, Apr. 2010.

[3] C. Chauvin, ''Human factors and maritime safety,'' *J. Navigat.*, vol. 64, no. 4, pp. 625–632, Sep. 2011.

[4] *Guidance on Fatigue Mitigation and Management*, Indian Med. Assoc., New Delhi, India, 2001.

[5] T. Chalder, G. Berelowitz, T. Pawlikowska, L. Watts, S. Wessely, D. Wright, and E. P. Wallace, "Development of a fatigue scale," *J. Psychosomatic Res.*, vol. 37, no. 2, pp. 147–153, 1993.
[6] M. R. Grech, A. Neal, G. Yeo, M. Humphreys, and S. Smith, "An examination of the relationship between workload and fatigue within and across consecutive days of work: Is the relationship static or dynamic?" *J. Occupational Health Psychol.*, vol. 14, no. 3, pp. 231–242, 2009.
[7] A. Sahayadhas, K. Sundaraj, and M. Murugappan, "Detecting driver drowsiness based on sensors: A review," *Sensors*, vol. 12, no. 12, pp. 16937–16953, Dec. 2012.
[8] K. Fujiwara, E. Abe, K. Kamata, C. Nakayama, Y. Suzuki, T. Yamakawa, T. Hiraoka, M. Kano, Y. Sumi, F. Masuda, M. Matsuo, and H. Kadotani, "Heart rate variability-based driver drowsiness detection and its validation with EEG," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 6, pp. 1769–1778, Jun. 2019.
[9] D. Qian, B. Wang, X. Qing, T. Zhang, Y. Zhang, X. Wang, and M. Nakamura, "Drowsiness detection by Bayesian-copula discriminant classifier based on EEG signals during daytime short nap," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 4, pp. 743–754, Apr. 2017.
[10] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognit. Lett.*, vol. 42, pp. 11–24, Jun. 2014.
[11] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, no. 4, pp. 917–963, Mar. 2019.
[12] S. Greenhill, S. Rana, S. Gupta, P. Vellanki, and S. Venkatesh, "Bayesian optimization for adaptive experimental design: A review," *IEEE Access*, vol. 8, pp. 13937–13948, 2020.
[13] D. R. Jones, "A taxonomy of global optimization methods based on response surfaces," *J. Global Optim.*, vol. 21, no. 4, pp. 345–383, 2001.
[14] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2546–2554.
[15] J. Mockus, "Application of Bayesian approach to numerical methods of global and stochastic optimization," *J. Global Optim.*, vol. 4, no. 4, pp. 347–365, Jun. 1994.
[16] A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter, "Fast Bayesian optimization of machine learning hyperparameters on large datasets," 2016, *arXiv:1605.07079*. [Online]. Available: http://arxiv.org/abs/1605.07079
[17] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*, vol. 55. North Chelmsford, MA, USA: Courier Corporation, 1965.
[18] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," 2010, *arXiv:1012.2599*. [Online]. Available: http://arxiv.org/abs/1012.2599
[19] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*. Berlin, Germany: Springer, 2003, pp. 63–71.
[20] B. P. Flannery, W. H. Press, S. A. Teukolsky, and W. Vetterling, *Numerical Recipes in C*, vol. 24. New York, NY, USA: Press Syndicate Univ., 1992, p. 78.
[21] J. Mockus, V. Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," *Towards Global Optim.*, vol. 2, nos. 117–129, p. 2, 1978.
[22] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," 2009, *arXiv:0912.3995*. [Online]. Available: http://arxiv.org/abs/0912.3995
[23] *Emotiv EPOC+*. Accessed: Jun. 17, 2019. [Online]. Available: https://www.emotiv.com/epoc/
[24] *Mysignals—Ehealth and Medical IoT Development Platform*. Accessed: Jun. 17, 2019. [Online]. Available: http://www.my-signals.com/
[25] *Norsk Senter for Forskningsdata as*. Accessed: Jun. 17, 2019. [Online]. Available: https://www.regjeringen.no/en/
[26] D. P. Subha, P. K. Joseph, R. Acharya, and C. M. Lim, "EEG signal analysis: A survey," *J. Med. Syst.*, vol. 34, no. 2, pp. 195–212, 2010.
[27] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, Feb. 2012.

THIAGO GABRIEL MONTEIRO received the B.Sc. degree in naval architecture and maritime engineering from the University of São Paulo, São Paulo, Brazil, in 2013, and the M.Sc. degree in ship design from the Norwegian University of Science and Technology (NTNU), Ålesund, Norway, in 2016, where he is currently pursuing the Ph.D. degree in physiological sensors fusion in the maritime domain.

His research interests include sensor fusion, machine learning, mental fatigue assessment, and human factors in maritime operations.

CHARLOTTE SKOURUP received the M.Sc. degree in mathematics and the Ph.D. degree in human–machine interaction from the Norwegian University of Science and Technology (NTNU), Trondheim, Norway, in 1994 and 1999, respectively.

She was an Associate Professor with the Department of engineering cybernetics, NTNU, from 2004 to 2015. She has been with ABB AS Oil, Gas and Chemicals, Oslo, Norway, since 2007, where she is currently the Section Manager of Products and Services R&D.

HOUXIANG ZHANG (Senior Member, IEEE) received the Ph.D. degree in mechanical and electronic engineering, in 2003, and the Habilitation degree in informatics from the University of Hamburg, in February 2011. Since 2004, he has been a Postdoctoral Fellow with the Department of Informatics, Faculty of Mathematics, Informatics and Natural Sciences, Institute of Technical Aspects of Multimodal Systems (TAMS), University of Hamburg, Germany. He joined the NTNU (before 2016, Aalesund University College), Norway, in April 2011, where he is currently a Professor in robotics and cybernetics. The focus of his research includes on two areas. One is on biological robots and modular robotics. The second focus is on virtual prototyping and maritime mechatronics. In these areas, he has published over 130 journal and conference papers and book chapters as the author or coauthor.