

The Realtime Assessment of Mental Workload by Means of Multiple Bio-Signals

Master thesis Report

Methodology and Statistics for the Behavioral, Biomedical and
Social Sciences

Utrecht University

Bart-Jan Boverhof, 6000142

Thesis Supervisor

Prof.dr.ir. B.P. Veldkamp

Date

January 11, 2020

1 Introduction

The topic of mental workload is a widely studied phenomenon across a variety of different fields, amongst others the field of ergonomics (Young, Brookhuis, Wickens, & Hancock, 2015), human factors (Pretorius & Cilliers, 2007) and neurosciences (Shuggi, Oh, She-wokis, & Gentili, 2017). A commonly utilized definition for mental workload, hereafter referred to as simply "workload", is the demand placed upon individuals whilst they carry out a particular task. As pointed out by De Waard and te Groningen (1996), the aforementioned definition is too simplistic, for it defines workload solely as an external phenomenon. Workload is, however, a person-specific construct, for the amount of perceived workload ushered by a given task may differ across individuals (De Waard & te Groningen, 1996). Hence, when referring to workload throughout this research, person-specific workload is implied specifically.

A commonly employed method for assessing workload is the well established NASA-Task Load Index questionnaire. This questionnaire inquires the respondent on the amount of perceived workload, and is constructed from six subjective sub-scales (Hart, 2006). Such an assessment is usually conducted post-experiment, which can in certain situations be deemed undesirable. Consider an experiment in which is aimed to assess workload of a pilot in flight. An evident approach towards such an experiment is that we wish to measure the degree of perceived workload during different phases of the flight. However, only after the simulation is concluded, a measurement in the form of a questionnaire can be administrated. In such a situation, utilizing a post-experiment assessment is prone to generate bias. A widely recognized bias is the observer-bias, advocating that participants in an experiment tend to overexaggerate the treatment effect when having to report it post-experiment (Mahtani, Spencer, Brassey, & Heneghan, 2018). In the light of the aforementioned example, pilots are expected to overexaggerate the degree of perceived workload when having to report it post-experiment.

An alternative approach to the assessment of workload is to collect physiological bio-signals during the experiment, and utilize these to classify workload. Examples of such bio-signals, hereafter referred to as "modalities", include techniques as electroencephalogram, eye-tracking, galvanic skin response, functional near-infrared spectroscopy etc. The advantage of such an approach is that complementary information streams, each stemming from a different modality, may all be interpreted simultaneously (Ramachandram

& Taylor, 2017). This has the potential of yielding a rich and multifaceted classification of construct such as workload. Additionally, it is possible to train a separate model for each individual, catering towards the individual perception of workload for that specific person. This approach, however, comes at the cost of an increase in complexity. This resides in the need to construct a complex framework that inputs the data from each of the utilized modalities, and ultimately outputs a single classification outcome.

The current research builds upon research conducted by Dolmans, Poel, van 't Klooster, and Veldkamp (in press), who proposed a deep-learning approach to multi-modular classification of workload. The current research differs from this previous endeavor in that it utilizes different modalities, and hence different data. In addition, the current research investigates upon the feasibility of a real-time approach. Real-time in this sense reflects the real-time classification of workload, i.e. the classification of workload whilst the experiment takes place. Doing so enables the possibility to conduct a dynamic experiment, the state of which can be altered by responding towards the classified degree of workload at a certain moment in time. Consider a simulation with the objective of educating its participant, such as a surgical simulation for educating surgeons-in-training. The learning experience of a single session could be dramatically enhanced when the state of the experiment is catered towards the individual learning process dynamically. For example, in case it is recognized early on in the simulation that a surgeon-in-training has difficulty with a specific procedure, the remainder of the experiment can be catered towards focusing on this specific procedure. By enhancing the learning experience in this way, the effectiveness of a single simulation session can be improved dramatically, entailing a more efficient learning process.

Three different modalities are utilized in the current research. Firstly the technique of electroencephalogram, hereafter referred to as "EEG", secondly the technique of galvanic skin response, hereafter referred to as "GSR" and thirdly the technique photoplethysmography, hereafter referred to as "PPG". It is of importance to recognize that the objective of the current research is not to gain insight into the most optimal model design for each of the previously delineated modalities. The objective is rather to construct a framework with which real-time classification can be managed, and to which modalities of choice can easily be added in future research endeavors. Consequently, one of two design principles on which the architecture of the framework reclines is the principle of modularity. Modularity refers to the extent to which different modalities can freely be added and/or removed

to the framework, without the necessity to re-architect and rebuild it entirely. The second design principle is the principle of generalizability, prescribing that the framework should not merely be utilizable in the context of workload, but also for the measurement of other mental constructs.

A deep-learning approach towards the construction of a real-time multi-modular framework is realized. Considering the complexity and sheer size of such a deep-neural network, an important point of attention is classification speed. The challenge in real-time classification with deep-learning is often not reaching adequate performance, but rather to attain adequate speed of classification. Deep neural networks easily constitute thousands of calculations to be made simultaneously, which is even more true for a multi-modular approach such as the current. In order for real-time classification to work, classification cannot take too long, for otherwise it is not real-time anymore and the previously delineated benefit of a real-time approach dissipates. As a consequence, multiple networks are considered and contrasted in terms of performance and ability to classify in real-time. Firstly, for each of the three modalities separately, a single-modular network is constructed. Secondly, a multi-modular network is architected. Additionally, several variations to this multi-modular framework are considered and contrasted. These variations are specified to differ in size, i.e. the amount of neurons and filters.

The objective of the current research is to explore the circumstances in which a multi-modular approach by means of deep learning is capable of real-time classification, whilst still ensuring ample and adequate performance. Ultimately, this line of research pursues the ability to conduct a dynamic experiment for multiple people simultaneously, and of which the state can be altered in real-time.

2 Methods

2.1 Related Work

The following section will provide an overview of preceding research on the most optimal network architecture for each single-modular network. Subsequently, the most feasible architecture for the multi-modular framework in its entirety will be explored, with particular attention on the data fusion strategy and a range model optimization techniques.

2.1.1 Electroencephalogram (EEG)

The first utilized modality is EEG, which constitutes a technique that detects electrical activity in the brain using electrodes. EEG is a commonly utilized method within the field of workload investigation. An overview of the complete literature on EEG applications with deep-learning was presented by Craik, He, and Contreras-Vidal (2019), who reported a total of 16 % of all available papers to cope with workload classification. With regards to these EEG applications, Craik et al. (2019) reported that studies mostly found deep belief networks and convolutional neural networks, hereafter referred to as "ConvNets", to perform best when classifying workload, and advice one of these approaches as a consequence.

Tabar and Halici (2016) proposed combining a ConvNet with a Stacked auto-encoder network, hereafter referred to as "SAE". The input layer was specified to feed into a convolutional layer with the objective of learning the filters and network parameters. The output of this convolutional layer was subsequently specified to feed into the SAE part, architected to include an input layer, 6 hidden layers and an output layer. A classification accuracy of 90 % was acquired with this network (Tabar & Halici, 2016).

Research by Schirrmeister et al. (2017) contrasted the performance of several ConvNets against the widely utilized baseline method for EEG classification, filter bank common spatial pattern, hereafter referred to as "FBCSP". A deep ConvNet, a shallow ConvNet, a deep-shallow hybrid ConvNet and a residual ConvNet were contrasted with an FBCSP. Both the deep and shallow ConvNets were found to reach at least similar, and in some regard better classification results as compared with the FBCSP baseline approach. Altogether, a deep ConvNet with four convolutional-max-pooling blocks was found to perform best, exhibiting an accuracy of 92.4 % (Schirrmeister et al., 2017).

2.1.2 Galvanic Skin Response (GSR)

The second utilized modality is GSR, measuring sweat gland on the hands and hereby inferring arousal. GSR activity is known to be significantly correlated with workload, as was demonstrated amongst others by Shi, Ruiz, Taib, Choi, and Chen (2007). As a consequence, GSR poses a widely utilized modality within the field of workload detection.

Sun, Hong, Li, and Ren (2019) explored the architecture for the most suitable deep-learning approach to the classification of several emotional states by means of GSR.

Several models were explored, amongst others a support vector machine, a ConvNet, a long-short-term-memory, hereafter referred to as "LSTM", network. Additionally, a hybrid model combining the ConvNet and LSTM approaches was explored. This aforementioned hybrid model was found to perform best, exhibiting an accuracy of 74% (Sun et al., 2019).

Dolmans et al. (in press) took a multi-modular approach to workload classification, and architected a variant on the previously delineated ConvNet-LSTM approach for the GSR modality specifically. The performance of this model was contrasted with a network consisting solely of fully connected dense layers. Conform with findings by Sun et al. (2019), the hybrid model was found to perform best, displaying an accuracy of 82 % (Dolmans et al., in press). The model architecture as utilized by Dolmans et al. (in press) deployed two convolutional max-pooling blocks and two LSTM layers.

2.1.3 Photoplethysmography (PPG)

The third modality constitutes PPG, which is a technique utilized to measure volumetric changes in blood in peripheral circulation. Zhang et al. (2018) contrasted several approaches towards workload classification, out of which PPG was found to perform among the best. As a consequence, PPG constitutes one of the most widely utilized approaches towards workload classification.

Work by Biswas et al. (2019) investigated upon a deep-learning approach towards PPG classification, with the objective to perform both bio-metric identification and obtain heart rate information. Exceptional results were attained with a deep neural network, attaining an average accuracy of 96 % (Biswas et al., 2019). This performance was realized with a ConvNet-LSTM hybrid, incorporating two convolutional max-pooling blocks followed with two LSTM layers.

2.1.4 Fusion Strategy

When taking a multi-modular deep-learning approach, information streams stemming from different modalities are required to be combined, i.e. "fused", at a certain point in the network in order to ultimately result in a single classification. Fusion can be done conforming different strategies. Several strategies as proposed by Ramachandram and Taylor (2017) have been considered.

Early, or data-level, fusion constitutes an approach that fuses data sources before

being fed into the network. Techniques that manage this include for example principle component analysis and factor analysis. Early fusing is usually proven to be challenging, residing in the fact that data streams stemming from different modalities often differ in their dimensionality and sampling rate. In addition, when endeavoring an early fusion approach, the oversimplified assumption of conditional independence is made implicitly. This assumption is unrealistic in nature, for data stemming from different modalities are expected to be correlated in practice (Ramachandram & Taylor, 2017).

Late, or decision-level, fusion refers to the process of aggregating the decisions of multiple networks, each architected and applied towards each modality separately. In case the data sources stemming from the various modalities are either correlated or ultimately inhibit a different dimensionality, late fusion is a much more feasible approach as contrasted with early-fusion (Ramachandram & Taylor, 2017).

Lastly, intermediate fusion is the most widely employed fusion strategy for multi-modular deep-learning problems. Data streams are usually fused by concatenation of the networks defined for each modality separately, followed by a higher order layer that ultimately results in a classification. This need not be a single layer, but could be multiple layers, as long as each modality ultimately feeds into the highest order output layer. The depth of the fusion, i.e. the specified number of fusion layers, can be chosen conform to the specific circumstances, posing intermediate fusion to be the most flexible and therefore the most widely adopted fusion strategy (Ramachandram & Taylor, 2017).

Indeed, when consulting the literature, one is forced to conclude that intermediate fusion strategies are the most prevailing for multi-modular deep neural networks. When taking such an approach, the higher order network architecture is required to be established, for which several previous endeavors will be considered. Rastgoo, Nakisa, Maire, Rakotonirainy, and Chandran (2019) utilize a multi-modular ConvNet approach, and fused the modalities by concatenation, followed with two LSTM layers, two dense layers and a softmax layer. A simpler approach is utilized by Han, Kwak, Oh, and Lee (2020), who utilized an intermediate fusion approach solely consisting of several fully connected dense layers, and ending with a soft-max layer. Lastly, Dolmans et al. (in press) took a relatively deep intermediate fusion approach, consisting of two dense layers, two convolutional layers followed by another two dense layers.

2.1.5 Model Optimization

The technique of batch normalization was originally proposed by Ioffe and Szegedy (2015), and is often applied in deep-learning with the objective of enhancing the stability of a network. Especially ConvNets capitalize on this technique. It is beneficial to include a batch normalization layer after each convolutional layer, re-centering and re-scaling the input feeding into subsequent layers. When incorporating a batch normalization layer, it is recommended to do so before feeding into the activation function (Ioffe & Szegedy, 2015). An increase in accuracy for EEG classification was attained by Dolmans et al. (in press) and Schirrmester et al. (2017) by specifying a batch normalization layer following each convolutional layer. Equally so, the best performing network for PPG data as proposed by Biswas et al. (2019) included a batch normalization layer succeeding each convolutional layer.

Pooling layers are commonly employed in ConvNets, often subceeding a convolutional layer with the purpose decreasing dimensionality. The objective of such layers are to merge similar features into one: for a more extensive elaboration see LeCun, Bengio, and Hinton (2015). Both Schirrmester et al. (2017) and Tabar and Halici (2016) specified a max-pooling layer after each convolutional layer within their EEG ConvNets. The network as proposed for GSR by Sun et al. (2019) incorporated a max-pooling layer after each but one of the convolutional layers. Lastly, the network as proposed for PPG by (Biswas et al., 2019) specified a max pooling layer after each convolutional layer.

Hyper-parameter optimization, hereafter referred to as "HPO", is a technique that can be used to optimize hyper-parameters such as learning rate, dropout probability and momentum. Substantial advancements within the deep learning community have been attained by utilizing HPO, especially with regards to ConvNet performance (Bergstra & Bengio, 2012). The Optuna toolbox provides a method for creating a parameter search space, from which values for the hyper-parameters can be sampled and optimization can be performed (Akiba, Sano, Yanase, Ohta, & Koyama, 2019).

2.2 Data

The subsequent section will provide an overview of the utilized data. Attention is primarily placed upon the experimental setup, the description of the respondents, the utilized devices and the synchronization process.






















2.2.1 Experimental Setup

The experimental setting for data collection is the open-source spaceship video-game Empty Epsilon, in which respondents are required to carry out tasks on a virtual spaceship (Daid & Nallath, 2016). This experiment was instituted by the Brain Computer Interfaces testbed lab, hosted by the University of Twente and carried out in cooperation with Thales group Hengelo. The experiment constituted three different segments, in each of which the respondent had to carry out tasks. These tasks were constructed to evoke varying degrees of perceived workload. Each segment consisted of six small sessions lasting roughly five to ten minutes. These sessions varied in difficulty, including two easy, two intermediate and two hard sessions per segment. A schematic overview of the experimental structure is depicted as figure 1.

After each small session, respondents filled in the TLX questionnaire comprising six questions each, resulting in 18 filled in questionnaires per respondent. Each questionnaire inquired upon the degree to which the respondent experienced workload during the preceding session. These ratings have been utilized to label the data for network training purposes. Within each segment, the order in which the sessions were presented was randomized. The order in which the three main segments were administrated was not randomized. Between every three sessions, respondents were requested to take a short two minute break.

The first segment emulated a scenario in which hostile spaceships approach the respondent’s spaceship. The respondent is required to quickly react by defusing the hostiles in order to survive. The increment in difficulty caused the process of defusing hostile spaceships to become more challenging, hereby aiming to cause an increase in workload. The second segment emulated a scenario in which the respondent had to navigate their spaceship through space, gathering as many way-points as possible. Obstacles around which the respondent had to carefully navigate and hostile spaceships the respondent had to decimate were introduced in the higher difficulty sessions. The third and final segment emulated a machine room, in which respondents had to control the power based on randomly generated requests. In the increased difficulty settings, variables that could overheat the spaceship were introduced, demanding the respondent to multi-task and aiming to increase workload.

Figure 1*Experimental setup*

Segment 1	Segment 2	Segment 3	
			Easy
			
			Intermediate
			
			Hard
			

2.2.2 Respondents

In total, 25 respondents have participated in the study. Currently, the data is still in the process of being collected, for which no additional descriptive statistics can be presented in this section as of yet. The respondents are students recruited from the University of Twente. Recruitment has been conducted with Sona, which is a cloud-based participant management system. A requirements for participation was that respondents didn't have any constraints that might interfere with the utilized sensors, such as for example a pacemaker. This was assessed by means of a short demographic questionnaire prior to the experiment. Additionally, the respondents were made aware of informed consent prior to the experiment with the objective to ensure completely voluntary participation. Respondents were able to draw back from the experiment at any time.

2.2.3 Devices and Sampling Rate

The Shimmer3 GSR+ sensor was used for both PPG and GSR measurements. The device is worn on the wrist, and is able to communicate both signals wirelessly. An ear-clip was utilized for measuring PPG, and converting this to estimate heart rate. Skin conductiv-

ity, or GSR, was monitored by two electrodes attached to the fingers (Shimmer-Research, n.d.). EEG measurement was conducted with the Muse 2, which is a multi-sensor head-band that provides feedback on brain activity (InteraXon, n.d.). The Shimmer3 GSR+ is able to read and output data signals on a sampling rate of 256 Hz, whereas the Muse 2 is able to sample at a maximum of 220 Hz.

As was set forth in the introduction, real-time classification requires a swiftly classifying network. A higher sampling rate equals more data traveling through the network, which decelerates classification speed. As a consequence, it is necessary to input data on the minimum required sampling rate with which key features can be detected a consistent manner, for each modality separately.

Fujita and Suzuki (2019) investigated the required sampling rate for PPG feature detection. The extent to which important features were detected was contrasted for several sampling rates. A sampling rate of 60 Hz was found to be the absolute minimum required sampling rate for extracting all commonly utilized features in a stable manner (Fujita & Suzuki, 2019). Utilizing a slightly higher sampling rate might be the safer option, however. The Shimmer3 GSR+ manufacturers recommend a sampling rate of 100 Hz for the PPG modality (Shimmer-Research, n.d.). Hence, a sampling rate of 100 Hz was specified for the PPG modality.

The required sampling rate for the GSR modality is substantially lower as compared with both PPG and EEG. In fact, the Shimmer3 GSR+ manufacturers recommend a sampling rate ranging between 0.03 and 5 Hz (Shimmer-Research, n.d.). A sampling rate of 5 Hz was specified as a consequence.

For the EEG modality, different features require a widely different sampling rate in order to be detected. Frequency bands for traditionally considered EEG features are defined on about 0.5-4 Hz for delta, and at most on about 16-24 Hz for beta. The gamma frequency band recently gained in popularity within the field of EEG however, and is defined on a frequency ranging up to 80 Hz (Weiergraeber, Papazoglou, Broich, & Mueller, 2016). In order to detect a feature residing on a 80 Hz frequency band, a substantially higher sampling rate is required to record the signal without aliasing. The required sampling rate can be determined by means of the Nyquist criterion for practical EEG sampling, defined as equation 1,

$$f_{samp} > 2.5 * f_{max} \quad (1)$$

where f_{samp} reflects the required sampling rate and f_{max} reflects the frequency range around which the feature to be detected resides (Srinivasan, Tucker, & Murias, 1998). Hence, in order to be able to detect the gamma frequency band, a sampling rate of $2.5 * 80 = 200$ was specified for the EEG modality. A summary of the specified sampling rates per modality are depicted in table 1.

Table 1

Sampling rate per modality

	Specified sampling rate (Hz)
Electroencephalogram (EEG)	200
Galvanic Skin Response (GSR)	5
Photoplethysmography (PPG)	100

2.2.4 Synchronization

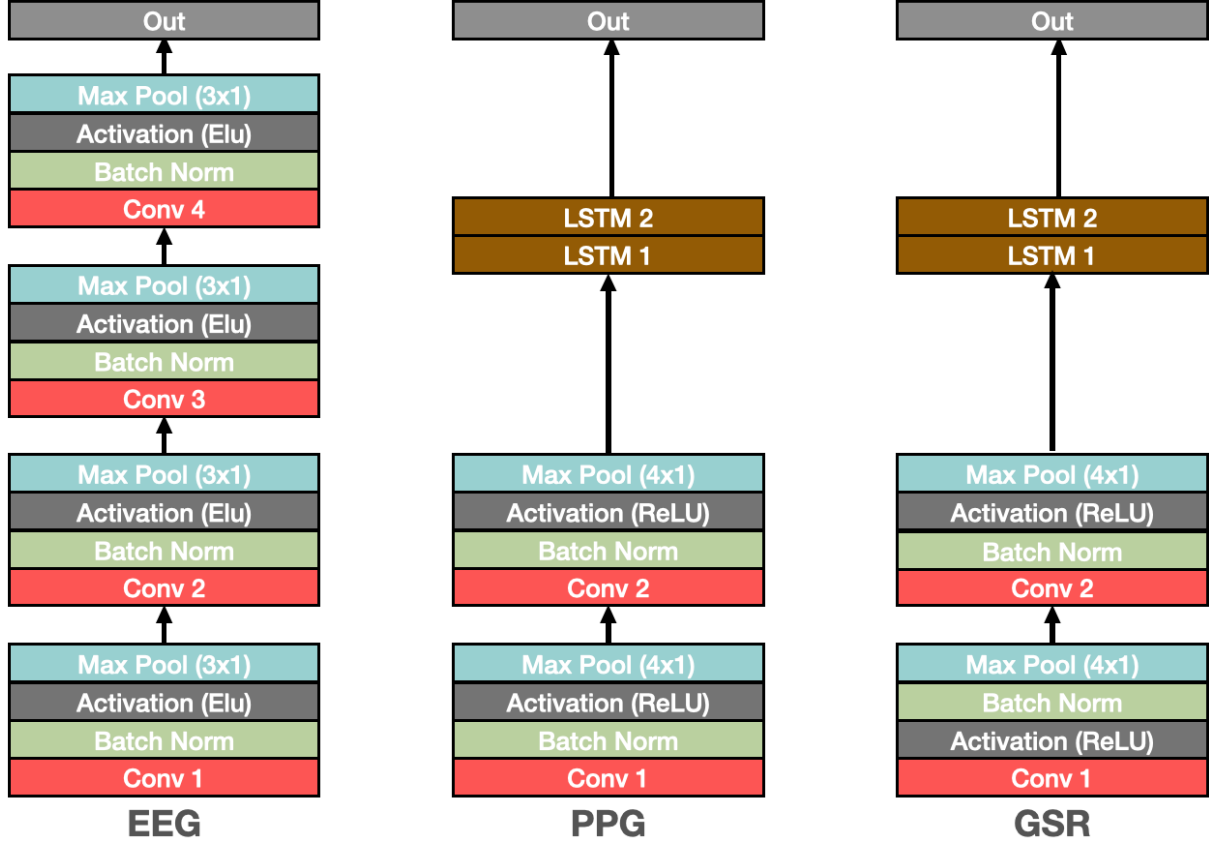
Data streams stemming from the different modalities were required to be properly synchronized. This was accomplished by means of an application called Lab-Streaming Layer, hereafter referred to as "LSL". The data streams stemming from the different modalities were all streamed to LSL during the experiment. LSL properly synchronized these data streams, such that they are parallel time-wise. Subsequently, all data was recorded into a single file per participant (Kothe, Medine, & Grivich, 2018).

2.3 Framework Architecture

As was elaborated on in the introduction, several networks have been compared in their ability to classify workload in real-time, and the performance with which this is managed. The upcoming section opens with the description of architecture of the three single-modular networks. Subsequently, the multi-modular network architecture will be elaborated on. Lastly, several variations made on this multi-modular architecture are discussed.

Figure 2

Three single-modular network architectures



2.3.1 Single-modular Network Architectures

The architecture of each of the single-modular networks is determined by combining insights from the literature. Each of the three networks merely utilized a single modality with which workload was classified. These networks and their architectures are depicted as figure 2.

The utilized network for the EEG modality was a ConvNet as proposed by Schirrmester et al. (2017). The network was designed to include four convolutional blocks, each constituting a convolutional layer, followed by a batch normalization layer. The Exponential Linear Unit, hereafter referred to as "ELU", function was utilized as activation function. Each convolutional block is closed with a max pooling layer of stride three.

The utilized network for the GSR modality was a LSTM ConvNet hybrid, inspired upon by the work of Sun et al. (2019) and Dolmans et al. (in press). The network was designed to include two convolutional blocks, each constituting a convolutional layer, followed by a batch normalization layer, the activation layer and closed with a max-pooling layer of stride four. The Rectified Linear Unit, hereafter referred to as "ReLU", function was utilized as activation function. Following these two convolutional blocks are two LSTM layers.

Lastly, the utilized network for the PPG modality was inspired from the network as proposed by Biswas et al. (2019). The network opens with two convolutional blocks, each consisting of a convolutional layer, batch normalization layer, activation layer and closed with a max pooling layer of stride four. The utilized activation function was the ReLU. Following these convolutional blocks are two LSTM layers, equal to the GSR network.

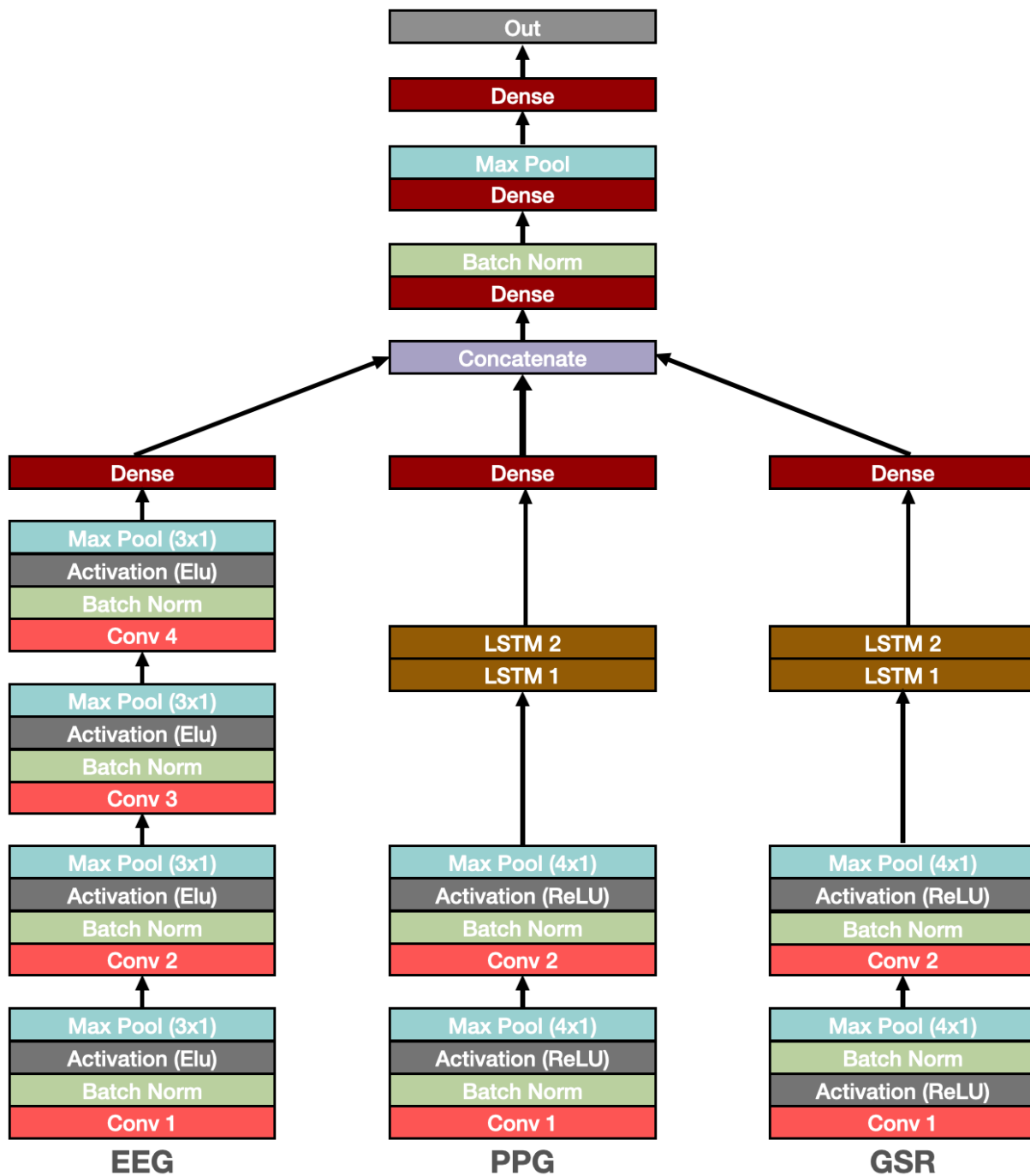
2.3.2 Multi-modular Network: Architecture

The network architecture that is utilized for the multi-modular approach is determined by a combination of the single-modular networks, as derived from the literature. The previously delineated design principles (i.e. the principles of modularity and generalizability) are taken into account when doing so. The visual representation of the multi-modular network is depicted as figure 3.

In order to combine the previously delineated single-modular networks, each of these distinct parts are closed with one fully connected dense layer before feeding into the head network. This is done in order to flatten all inputs into a lower dimensional space, such that concatenation is possible. The head network consists of four dense layers. These layers are alternated with a batch normalization and max-pooling layer with the objective of stabilization.

2.3.3 Multi-modular Network: Variations

Speed is a potential bottleneck for a multi-modular network that should be able to classify in real-time. The previously delineated network is substantially complex in nature. Therefore, several variations of this network have been investigated upon, each differing in their complexity. These variations are not made by altering the network architecture, for deviating from the distinguished architecture might be detrimental towards performance.

Figure 3*Multi-modular Network Architecture*

The goal is to propose a network that is fast enough for real-time classification, whilst maintaining the highest amount of accuracy as possible. Three different variations with regards to size of the network as depicted in figure 3 have therefore been considered.

Network size can be understood as the amount of specified filters for convolutional layers, and the amount of specified neurons for all other utilized layers. A decrease in the amount of filters and neurons constitutes a decrease in network size, and consequently a decrease in the amount of required calculations. Doing so brings about an increase in speed. An overview of all three multi-modular network variations, and the amount of specified neurons/filters per layer, is provided in table 2. Network 1 is referred to as the full network, and is the biggest in terms of size. The size of network 2 constitutes of 75 % of the size of the full network. Network 3 constitutes of 50 % of the size of the full network.

Table 2*Model variation sizes*

	EEG	GSR	PPG	Head
Network 1	Conv1: 25	Conv1: 128	Conv1: 128	Dense: 712
	Conv2: 50	Conv2: 128	Conv2: 128	Dense: 356
	Conv3: 100	LSTM1: 256	LSTM1: 256	Dense: 178
	Conv4: 200	LSTM1: 256	LSTM2: 256	
	Dense: 200	Dense: 256	Dense: 256	
Network 2	Conv1: 18	Conv1: 96	Conv1: 96	Dense: 534
	Conv2: 34	Conv2: 96	Conv2: 96	Dense: 267
	Conv3: 75	LSTM1: 192	LSTM1: 192	Dense: 134
	Conv4: 150	LSTM1: 192	LSTM1: 192	
	Dense: 150	Dense: 192	Dense: 192	
Network 3	Conv1: 13	Conv1: 64	Conv1: 64	Dense: 356
	Conv2: 25	Conv2: 64	Conv2: 64	Dense: 178
	Conv3: 50	LSTM1: 128	LSTM1: 128	Dense: 89
	Conv4: 100	LSTM1: 128	LSTM2: 128	
	Dense: 100	Dense: 128	Dense: 128	

Note: For all convolutional layers the depicted number reflects the amount of utilized filters, whereas for LSTM layers it reflects the amount of nodes.

2.4 Model Evaluation

The network performances have been assessed and contrasted by means of several performance metrics. The utilized metrics constitute six well known and widely applied metrics, all constructed from the confusion matrix, depicted as table 3.

Table 3

Confusion matrix

	True Positive	True Negative
Predicted Positive	a	b
Predicted Negative	c	d

The measures accuracy, sensitivity, specificity, PPV, NPV and F1 have been utilized in order to asses network performance. The network that performs best across these measures was considered to be the superior performing network. Table 4 depicts the constitution of these performance metrics, by partly referring to confusion matrix depicted as table 3.

Table 4

Performance Metrics

Accuracy:	$\frac{a+d}{a+b+c+d}$
Sensitivity:	$\frac{a}{a+c}$
Specificity:	$\frac{d}{b+d}$
Positive Predicted Value (PPV):	$\frac{a}{a+b}$
Negative Predicted Value:	$\frac{d}{c+d}$
F1-measure:	$\frac{2*Sensitivity*PPV}{Sensitivity+PPV}$

References

- Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining* (pp. 2623–2631).
- Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1), 281–305.
- Biswas, D., Everson, L., Liu, M., Panwar, M., Verhoef, B.-E., Patki, S., . . . others (2019). Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment. *IEEE transactions on biomedical circuits and systems*, 13(2), 282–291.
- Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of neural engineering*, 16(3), 031001.
- Daid, & Nallath. (2016). *Empty epsilon multiplayer spaceship bridge simulation*. URL: <https://github.com/daid/EmptyEpsilon>. GitHub.
- De Waard, D., & te Groningen, R. (1996). *The measurement of drivers’ mental workload*. Groningen University, Traffic Research Center Netherlands.
- Dolmans, T., Poel, M., van ’t Klooster, J.-W., & Veldkamp, B. (in press). Percieved mental workload detection using intermediate fusion multi-modal networks.
- Fujita, D., & Suzuki, A. (2019). Evaluation of the possible use of ppg waveform features measured at low sampling rate. *IEEE Access*, 7, 58361–58367.
- Han, S.-Y., Kwak, N.-S., Oh, T., & Lee, S.-W. (2020). Classification of pilots’ mental states using a multimodal deep learning network. *Biocybernetics and Biomedical Engineering*, 40(1), 324–336.
- Hart, S. G. (2006). Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 50, pp. 904–908).
- InteraXon. (n.d.). *Muse 2 brainwave activity headband*. Retrieved from <https://choosemuse.com/muse-2/>
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.
- Kothe, C., Medine, D., & Grivich, M. (2018). Lab streaming layer (2014). URL: <https://github.com/scn/labstreaminglayer> (visited on 02/01/2019).

- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436–444.
- Mahtani, K., Spencer, E. A., Brassey, J., & Heneghan, C. (2018). Catalogue of bias: observer bias. *BMJ Evidence-Based Medicine*, 23(1), 23.
- Pretorius, A., & Cilliers, P. (2007). Development of a mental workload index: A systems approach. *Ergonomics*, 50(9), 1503–1515.
- Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Processing Magazine*, 34(6), 96–108.
- Rastgoo, M. N., Nakisa, B., Maire, F., Rakotonirainy, A., & Chandran, V. (2019). Automatic driver stress level classification using multimodal deep learning. *Expert Systems with Applications*, 138, 112793.
- Schirrmeister, R. T., Springenberg, J. T., Fiederer, L. D. J., Glasstetter, M., Eggensperger, K., Tangermann, M., ... Ball, T. (2017). Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11), 5391–5420.
- Shi, Y., Ruiz, N., Taib, R., Choi, E., & Chen, F. (2007). Galvanic skin response (gsr) as an index of cognitive load. In *Chi'07 extended abstracts on human factors in computing systems* (pp. 2651–2656).
- Shimmer-Research. (n.d.). *Shimmer3 gsr unit*. Retrieved from <https://www.shimmersensing.com/products/shimmer3-wireless-gsr-sensor>
- Shuggi, I. M., Oh, H., Shewokis, P. A., & Gentili, R. J. (2017). Mental workload and motor performance dynamics during practice of reaching movements under various levels of task difficulty. *Neuroscience*, 360, 166–179.
- Srinivasan, R., Tucker, D. M., & Murias, M. (1998). Estimating the spatial nyquist of the human eeg. *Behavior Research Methods, Instruments, & Computers*, 30(1), 8–19.
- Sun, X., Hong, T., Li, C., & Ren, F. (2019). Hybrid spatiotemporal models for sentiment classification via galvanic skin response. *Neurocomputing*, 358, 385–400.
- Tabar, Y. R., & Halici, U. (2016). A novel deep learning approach for classification of eeg motor imagery signals. *Journal of neural engineering*, 14(1), 016003.
- Weiergraeber, M., Papazoglou, A., Broich, K., & Mueller, R. (2016). Sampling rate, signal bandwidth and related pitfalls in eeg analysis. *Journal of neuroscience methods*, 268, 53–55.
- Young, M. S., Brookhuis, K. A., Wickens, C. D., & Hancock, P. A. (2015). State of science: mental workload in ergonomics. *Ergonomics*, 58(1), 1–17.

Zhang, X., Lyu, Y., Hu, X., Hu, Z., Shi, Y., & Yin, H. (2018). Evaluating photoplethysmogram as a real-time cognitive load assessment during game playing. *International Journal of Human-Computer Interaction*, 34(8), 695–706.