HP Omnicept Cognitive Load Database (HPO-CLD) – Developing a Multimodal Inference Engine for Detecting Real-time Mental Workload in VR

The HP Omnicept cognitive load database was created as part of an effort to reliably estimate users' mental effort (i.e., cognitive load) in real-time while they completed cognitively demanding tasks in virtual reality (VR) environments. In this technical report, we present a novel and robust machine learning method for assessing cognitive workload based on behavioral and physiological measures. Over approximately one hour, participants completed multiple tasks, requiring different levels of mental effort while we passively recorded their physiology, tracked their performance, and collected self-reports. We used these data to train machine learning models to predict task difficulty and participants' momentary self-reported cognitive load. Using a novel labeling pipeline, we achieved an average classification accuracy of 79.08% with a mean absolute error of 0.1106. These results indicate that, with a combination of behavioral and physiological indicators, we can reliably predict cognitive load in real-time, without calibration. As part of this white paper, we are releasing our test dataset (n = 100) for use by researchers and developers interested in machine learning, virtual reality, learning & memory, cognition, or psychophysiology. This dataset includes recordings from multiple sensors (including pupillometry, eye-tracking, and pulse plethysmography), self-reported cognitive effort, behavioral task performance, and demographic information on the sample.

Full technical report and downloadable dataset available April 30, 2021

Siegel, E.H., Wei, J., Gomes, A., Oliviera, M., Sundaramoorthy, P., Smathers, K., Vankipuram, M., Ghosh, S., Horii, H., Bailenson, J., & Ballagas, R. "HP Omnicept Cognitive Load Database (HPO-CLD) – Developing a Multimodal Inference Engine for Detecting Real-time Mental Workload in VR" *Technical Report*, HP Labs, Palo Alto, CA. Available at:
https://developers.hp.com/omnicept/omnicept-open-data-set-abstract

1. Introduction

1.1 Cognitive load. In scientific terms, the amount of mental effort required to perform a task or learn something new, often called cognitive load, and has been studied by researchers interested in learning and performance for over a century[1]–[6]. Every person has their own information processing capacity (also called working memory capacity or short-term memory), and it is fixed (unchanging), [7], limited (small capacity) [8], [9], and varies from person to person [10], [11]. Information processing capacity can be thought of like the amount of food a person can hold in their mouth at one time. Sure, it may be possible to fit 44 marshmallows in your mouth at once, but is it comfortable? Probably not. Is it an optimal way to eat? Definitely not. We can think of information processing capacity in a similar same way. Too much information to process at once and you cannot think; too little and it is not worth the effort.

In fact, research shows, that cognitive load is an important predictor of learning, memory, performance, stress, and burnout [12]–[16]. An examination of the theoretical underpinnings of cognitive load, particularly cognitive load theory[1], [2], can provide insight into why. Cognitive load theory suggests that successful completion of any task (large or small) relies on the complex interplay between sensory inputs, long-term memory (acting as a repository of previously acquired knowledge and skills) and working memory. Working memory acts as an intermediate state between sensory and long-term memory, attaching meaning to the sensory information by integrating newly learned information into longer-term memory. Both sensory and long-term memories have flexible capacities and are capable of processing large volumes of information. Working-memory, on the other hand, is comparatively limited [7], [10]. Attention manages the function of working memory by guiding it to relevant sensory information and stored knowledge, thereby directing the learning process and increasing (or decreasing) the efficiency of working memory[11]. An individual's cognitive load in each moment is an amalgam of these attentional, sensory, and memory processes. When we measure cognitive load, we are estimating the amount of mental resources being utilized to complete the task at hand.

1.2 Indicators of cognitive load Given the importance of mental effort for understanding human cognition, researchers have long sought reliable, objective indicators of cognitive load (i.e., measures that do not require asking participants about their experience). Some researchers have attempted to estimate cognitive load unobtrusively by recording and categorizing participants' facial expressions during the task (e.g., [17]–[19]). Unfortunately, facial expressions have been unreliable indicators for both practical (dynamic facial expressions can be difficult to infer, [20]) and theoretical (the categories themselves are up for debate, [21]) reasons. Researchers have explored acoustic features of the voice (e.g., [22]) to predict cognitive load. However, findings do not reliably generalize across cognitive load tasks [23], and people are not always speaking while engaging in tasks

More reliable, and generalizable, indicators of mental effort are measures of the peripheral nervous system and tracking the behavior of the eyes [24]. Cardiovascular activity (e.g., blood pressure, [25]; heart rate, [26]–[29].; and high frequency heart rate variability, [29], [30]), the electrical conductance of the skin, called electrodermal activity (EDA) or skin conductance [31],

[32], and the dilation of the pupil, called pupillometry, have proven reliable in estimating changes in cognitive load levels [16]. For example, multiple studies have shown a strong relationship between task demands and pupil dilation (e.g., [18], [33]).

1.3 Review of past ML (Machine Learning) research. Physiological indicators that are sensitive to cognitive load provide the opportunity to develop non-contact approaches to estimate cognitive load in real time. Researchers have developed several well-known machine learning algorithms to predict cognitive load using features extracted from different signals.

The algorithms used most in this domain include k-nearest neighbor (KNN), naïve bayes (NB), logistic regression, linear discriminant analysis (LDA), support vector machine (SVM), ensemble methods, (e.g. random forest and XGBoost), and neural networks. These machine learning models are trained to predict a users' cognitive load based on physiological features from one or multiple signal modalities. For example, Nourbakhsh et al. [34], used SVM and NB to train a cognitive load prediction model based on skin conductance and blink features. Haapalainen et. al [26] trained NB to classify three levels of cognitive load using electrocardiogram (ECG) and eye movement features. Ferreira et. al [29] used quadratic discriminant analysis for cognitive load classification with ECG, EEG and GSR features. Jimenez-Molina et. al., [35] employed logistic regression, SVM, and neural network models based on features from EDA, ECG, PPG, EEG, temperature, and pupil dilation signals. Ahmed et. al [36] used SVM and kNN to predict cognitive load with heart rate variability features derived from an ECG signal (for more see, [37]).

Both cognitive load and psychophysiological research suggest that different signal modalities can carry complementary and overlapping information. For example, as a subject experiences more cognitive load, the heart may respond with accelerated heart rate but lower heart rate variability [24], [26], [27], [36], while the eyes may respond with a pupil widening [16], [38], [39], more saccades [18], [40], [41] or increased blink activity [42]. For this reason, the fusion of multi-modal information has the potential to improve cognitive load prediction accuracy because the correlation of these different physiological responses across the body gives us a clearer picture of the autonomous nervous response than any signal individually. In addition, multi-modal fusion increases the robustness of the model by minimizing the effects of noise in any individual channel and by correlating physiological responses across the body that may have the same underlying cause (e.g., increased activity of the sympathetic nervous system) but unfold on different timescales (e.g., sympathetically mediated changes in pupil size happen more quickly than changes to pulse transit time).

In machine learning, common multi-modal fusion strategies include feature level fusion (i.e., early fusion), decision level fusion (i.e., late fusion), and hybrid fusion. The work in [29, 35, 36] applied feature level fusion, which simply composed features from different signals together to best exploit the interactions among multiple modalities. Alternatively, Islam et. al. [43], [44]proposed using shared information to merge features from different input modalities to predict drivers' mental workloads. In [45]Zhang et. al. explored using decision fusion, which combined sub-decisions of models trained on each modality, to achieve a more robust prediction. Zhang et. al. has also experimented with using hybrid fusion, a combination of feature fusion and

decision fusion, to improve the performance of cognitive load inference. In recent years, due to the success of applying deep neural networks in computer vision and natural language processing, neural network models have also shown promise for predicting cognitive load. Sarkar et al. [46], developed a multitask, deep neural network to classify high and low cognitive loads using ECG signals. Saha et. al. [47]introduced a long short-term memory (LSTM) followed by a multi-layer perception (MLP) algorithm to classify cognitive loads based on EEG signals. These studies, and others suggest that the deep learning models perform significantly better than the classical models, such as kNN, SVM, and LDA.

1.4 Overview of Current Work. This study was conducted as part of a larger, international research effort to develop a commercial, AI "inference engine" to recognize and assess real-time mental effort (i.e., cognitive load) in virtual reality (VR). Our goal was to develop a scientifically validated solution that could reliably predict cognitive load in the general population. Because of this, our approach differed from past research and includes important innovations from other commercially available "cognitive workload" solutions. First, our sample is larger and more diverse than previous studies. We did not exclude participants, nor did we purposefully select a homogeneous population to minimize variability from individual differences. It was critical for our models to represent the full population of potential users and research suggests that sampling bias is one of the most pervasive forms of bias in AI [48], [49]. Second, our feature selection process focused on robustness verses parsimony. It was more important that we could reliably predict cognitive load under a variety of user and task conditions (e.g., a task that impedes eye movement, or a user with cardiac arrhythmia) than that we identified a small number of particularly impactful features. Third, we wanted a real-time, calibration-free solution, which required important innovations in signal processing, labeling, and feature engineering. In general, commercial AI models that predict mental workload do so post-hoc, during some type of after-action-review (e.g., a user's cognitive load is estimated by subtracting responses during a task from some type of baseline or calibration task). Finally, we wanted to share a subset of our dataset with the larger scientific community for validation of our models and to seed new research ideas. We hope learning, psychophysiology, cognitive, education, and virtual reality researchers will find these data useful and fruitful.

2. Method

2.1 Participants. 738 participants were recruited from various communities for this experiment. The age range of the participants was between 19 and 61. Participants did not report ophthalmological conditions (other than corrected vision). Participants received a payment in local currency as remuneration for participating in the study. All participants received detailed information about the nature of the study, their role, and how their data would be managed, stored, and used to develop new products, and published as a part of this open dataset. Each participant gave their informed consent.

2.2 Design and Procedure. Participants completed a series of tasks presented in a randomized order, designed to require different levels of mental effort, or cognitive load (CL), to complete. Three CL levels were manipulated (low, medium, high), and each level was repeated

three times (3 low trials, 3 medium trials, 3 high trials), in a random order, for a total of nine trials (Figure 1). At the end of each trial, participants rated how mentally demanding they found the task (Figure 1).

The series of tasks were designed to stimulate cognitive load using a multitasking paradigm, spanning multiple modalities, to best approximate a wide range of cognitive load tasks in the wild. In a low CL trial, participants completed a visual vigilance task in which five balls appear in the scene. The ball to be tracked is briefly highlighted in a different color. The dots then move in a random, diverging pattern around the screen. The balls then move around randomly and eventually settle in one of five spots on the screen (labelled A through E, see Figure 1A). The users need to indicate where the target ball landed.

In the medium CL trial, participants completed the same visual vigilance task while also performing an arithmetic task. Numbers would progressively be presented to the screen while the balls were in motion (see Figure 1B). The numbers would disappear when prompted for a response, which forced the users to perform the arithmetic task in parallel with the visual vigilance task.

In the high CL trial, a third audio vigilance task was added (see Figure 1C). In this condition, the subject is listening for an audio beep. An additional visual element of a spinning wheel is layered into the scene, and the direction of spinning randomly changes throughout the task. When the audio prompt occurs, the subject should indicate using the corresponding controller trigger which direction the green wheel is spinning. Note that the CL tasks are designed such that each level of task difficulty is objectively more difficult as the more difficult conditions completely incorporate the easier conditions.

Participants began the task with several practice trials to introduce them to the procedure. The practice trials started with a low CL trial. If participants failed to complete the task successfully on the first practice trial, they were given two more tries to complete the task. If successful, participants moved on to medium CL, followed by high CL practice trials. If a participant failed to successfully complete a practice trial after three tries, they skipped to the next trial. This feature was added ensure that participants had relatively equal exposure to the stimuli and were in VR for roughly the same amount of time (minimizing effects from fatigue and/or VR sickness).

In the testing phase, participants were shown a total of nine tasks (three low, three medium and three high) in a randomized order to minimize carry over effects from the last trial. Because participants were, at this point, familiar with the stimuli and virtual reality, we expect that cognitive load effects from the testing phase are the result of our manipulations.

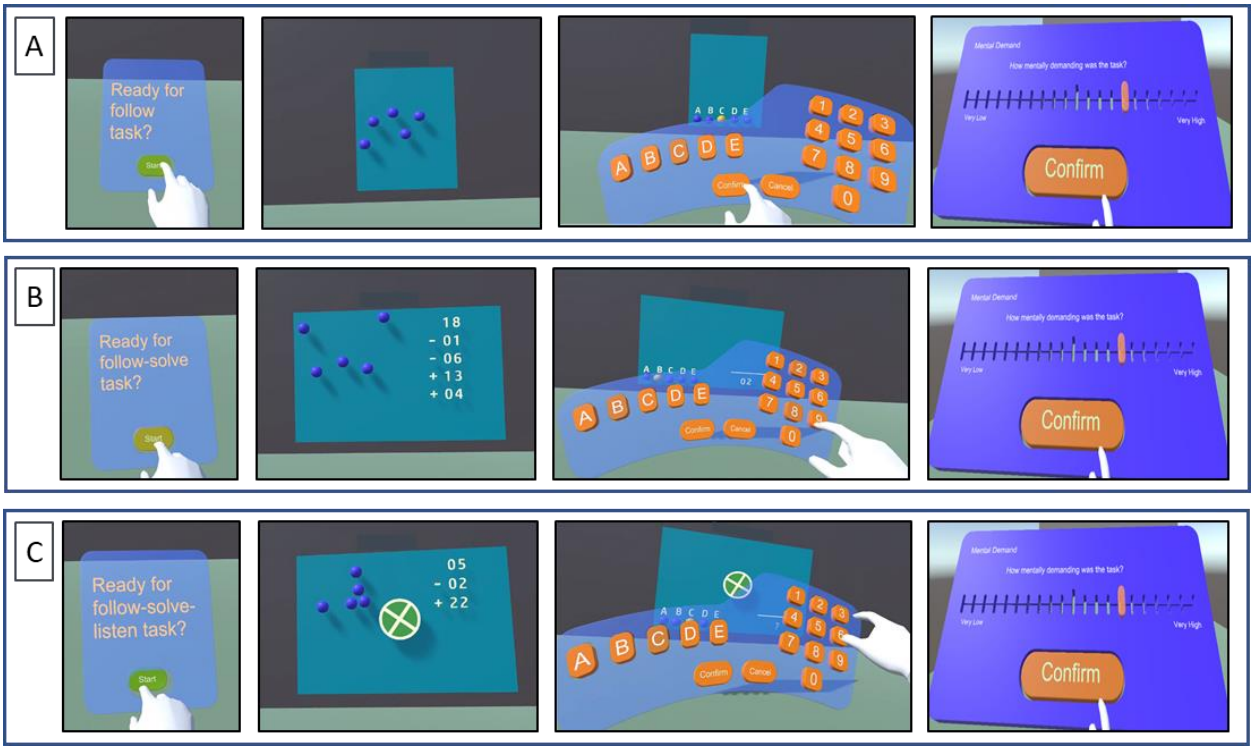

Figure 1. Example stimuli from the cognitive load task. On a given trial, participants complete tasks designed to induce different levels of mental effort (low, medium, and high). Box A is an overview of a low mental effort trial. In a low trial, five dots move in a random, diverging pattern around the screen. Participants' task is to track a single dot and, when the dots stop moving, indicate which of dots they were tracking. Box B is an example of a medium mental effort trial in which participants complete the dot tracking task and a mental math task where numbers and operators appear at random intervals and they are instructed to do the math in their heads. When the dots stop moving and the numbers stop appearing on the screen, participants report which dot they were tracking and their answer to the mental math problem. Box C is an example of a high mental effort trial in which participants complete the dot tracking, mental math, and a third, vigilance task, in which they monitor a spinning wheel that changes directions. When they hear a beep, the wheel stops spinning, and they report which direction the wheel was spinning (while simultaneously tracking the dots and doing the math problem). In all three conditions, at the end of each trial, participants then self-report how mentally demanding they found the task on a continuous rating scale from very low to very high.

2.3 Setup and Materials

Participants completed a VR experience that was designed to stimulate different levels of cognitive load while we passively recorded data from multiple sensors. The experience was developed by the authors in Unity3D and included a series of tasks similar to the desktop-based stimulus media proposed by Bartels & Marshall [50], but modified for optimization in VR. Participants completed the experience using the HTC Vive Pro-eye head mounted display with dual OLED 3.5" diagonal screen, 1440 x 1600 (pixel per eye) resolution, 90Hz refresh rate, 110° field-of-view. The Vive ProEye includes Tobii eye tracking (120Hz gaze output) and pupillometry capabilities[1]. To measure cardiac activity, we used a BITalino (r)evolution wired

[1] For https://www.vive.com/us/product/vive-pro-eye/specs/.

pulse plethysmography (PPG) sensor (bitalino.com). PPG was collected from the forehead, using a PPG sensor affixed directly to the mask (Figure 2, Box A) and from the finger (Figure 2, Box B).

Figure 2. Sensor and HMD set up for data collection. Box A depicts a Vive Pro-Eye head mounted display with eye-tracking and pupillometry capabilities. Our version was modified to include a pulse plethysmography sensor. As you can see in this image, the PPG sensor was affixed to the top of the mask in between the lenses. We also recorded PPG data from the finger

During the task, we collected performance from each task and, at the end of each trial, subjective ratings of cognitive load (Figure 1). Using a modified version of the NASA Task Load Index (TLX), participants rated how "mentally demanding" the found the task on a continuous scale from very low to very high [51].

3. Data Processing and Feature Extraction:

We measured dilation of the pupil and tracked participants' eyes using a Tobii Eye Tracking system integrated in the HTC Vive Pro-Eye (https://vr.tobii.com/integrations/htc-vive-pro-eye/). From Tobii's API, we were able to collect data on pupil position, pupil diameter, gaze position, and gaze direction from participants as they worked on the CL tasks. We focused

primarily on pupillometry and rapid movements of the eye between points (i.e., saccades) because research suggests that these indicators are particularly sensitive to changes in cognitive load and relatively context independent. Gaze data, on the other hand, is determined largely by the content presented (i.e., it is situation specific) and is not, itself, a reliable indicator of cognitive load. The Tobii output signals can be regarded as a multivariate time series that captures the values of variables over time. Table 1 includes a full list of features, the sensor, and the feature family (e.g., pupil features, fixation features, or heart rate variability features)

3.1 Eye Tracking. We used the multivariate time series (MTS) of six variables (dimensions), three for each eye (pupil diameter, pupil position on the x-axis, and pupil position on the y-axis). The procedure we used to process these signals and extract features is outlined below:

1. Data Buffering

Streamed data from the six-dimensions of Tobii data was cached for processing and feature extraction in 12.5 second windows.

2. Signal Normalization

Every 12.5 seconds, minimum, maximum, mean, and standard deviation for each of the six dimensions was calculated and updated to reflect the current time stamp. These summary statistics from previous windows were then used to normalize the data in real-time, resulting in a rolling normalization that updated every 12.5 seconds.

3. Blink Detection, Blink Features Extraction, and Blink Removal

To detect and remove blinks from pupil data, we used guidance and techniques developed by Mathôt, et al. [52] and others to identify and reconstruct the signal loss from closing the eyes during blinks. To start, we normalized pupil diameter signals to calculate standard deviation in rolling windows. If, during a given time interval, the standard deviation of pupil diameter was greater than a threshold of 0.4, we marked these as blink events. Once we identified the blink events, we calculated the longest blink duration, mean blink duration and blink rate (in Hz) and used these as our blink features. Finally, reconstructed the pupil data during a blink, by removing the blink events and filling in the missing data with interpolation.

4. Pupil Diameter Feature Extraction

After blink removal and data interpolation, pupil diameter features were extracted by calculating the mean and standard deviation of the normalized pupil diameter data.

5. Frequency Domain Features Extraction

Following blink removal and interpolation, we used the welch function from the Python scipy.signal library to calculate power spectral density features within specified frequency bands. The spectral features include avg_power within [0, 0.2] Hz, avg_power within [0.2, 0.4] Hz, avg_power within [0.4, 0.6] Hz, avg_power within [0.6, 1] Hz.

6. Wavelet Features Extraction

Research suggests that time-frequency decomposition techniques (wavelets) can be a powerful tool for estimating cognitive load (e.g., [38], [39]). To extract wavelet features we applied low-pass filtering on the pupil diameter signals. Next, we used the Python wavedec function from the pywt package to calculate wavelet features (e.g., rbio3.1, twitch sum, haar, twitch sum, rbio6.8, twitch frq)

7. Saccade Features, Saccade Speed Features, Fixation Features, and Saccade/Fixation Ratio Features Extraction

Saccades are characterized by a quick, simultaneous movement of both eyes in the same direction and can be detected by calculating rapid changes in gaze direction. To calculate saccades in our data, we estimated the speed of gaze movement using pupil position data (e.g. left_pupil_position_in_tracking_area_x, and left_pupil_position_in_tracking_area_y). Similar to blink detection, we started by selecting time intervals when the speed of gaze movement exceeded a threshold. For all saccade events within the data buffer, we calculate statistical features of each saccade, such as number of saccades within the buffer, longest saccade, mean and standard deviation of the duration of saccades (in seconds), and saccade rate (in Hz). We also calculate statistical features that describe the speed of eye movement, such as maximum speed, mean and standard deviation of gaze movement speed, gaze movement path length, total saccade duration (second), and saccade rate (Hz).

We extract fixation events by selecting time intervals when changes in gaze direction is below the threshold. For all fixation events within the data buffer, we calculate statistical features, such as number of fixations, longest fixation, and mean and standard deviation of fixation durations (in second), and fixation rate (in Hz).

Finally, we calculate the ratio of saccade features to fixation features, such as the ratio of fixation rate over saccade rate, ratio of longest fixation duration over longest saccade duration, ratio of sum fixation duration over sum saccade duration, and ratio of mean fixation duration over mean saccade duration.

8. Feature Normalization

The global minimum, maximum, mean, and standard deviation values of all the extracted features for each individual were tracked and updated every 12.5 seconds (data buffer) from the start of recording until the end of the recording. All extracted features were normalized using the updated global statistics.

3.2 Pulse Plethysmography (PPG). PPG is a light-based sensor that can measure heart activity by detecting changes in blood flow at the location where the sensor is applied to the skin (e.g., blood oxygenation and volume). In this data collection experiment, we used a Bitalino (r)evolution (http://bitalino.com) sensor board, with a PPG sensor embedded in the mask and one applied to the fingertip of participants (Figure 1). PPG sensors are a reliable and relatively non-invasive way to estimate cardiac changes but the signal is particularly prone to artifacts and other

sources of noise. For example, good contact between the sensor and the skin is necessary for reliable readings, making PPG sensitive to changes in movement. More complex sources of noise can occur because PPG measures the heart from pulses recorded in the periphery (arms, wrists, foreheads, etc.) leading to variability in the time it takes the pulse to travel to the recording sight (called pulse transit time, or PTT). PTT can differ depending on predictable internal characteristics like a participants' height, age, or the health of their vasculature but PTT also varies with increased (or decreased) physiological arousal in complex, difficult to measure ways. To account for these sources of variability in the PPG signal, we developed a six-step processing algorithm to extract features known to be related to tracking and estimating cognitive load.

1. Filter Data

We used a bandpass FIR (finite impulse response) filter with a lower cutoff frequency of 0.5Hz and an upper cutoff frequency of 5Hz. In the resulting signal sources of noise outside of the filter range are removed.

2. Seasonal decomposition

We deconstructed the signal using seasonal decomposition, which decomposes the signal based on rates of change. This allowed us to remove slower moving component series in the PPG signal, returning a detrended PPG signal. The seasonal decomposition was computed using the statsmodels package in python (statsmodels.org).

3. Identify peaks

As a first pass, our peak detection algorithm uses the scipy.signal.find_peaks algorithm, with a specified distance corresponding to a maximum of 140 beats per minute. Under some circumstances, like intense aerobic exercise, it is possible to exceed 140 beats per minute, most scenarios for our target audience will not exceed this threshold.

4. Filter Peaks based on skewness

We do not expect all peaks identified in the first pass to be valid peaks. Elgendi [53] demonstrated that skewness of the photoplethysmogram (PPG) signals is the optimal method for determining the quality of the signal and can be formalized into a signal quality index. Skewness is a measure of symmetry in the PPG signal.

Characteristic PPG signals have a skewed shape that can be attributed to the systolic and diastolic peaks of the signal. Because the systolic peak always occurs prior to the diastolic peak, in a tight time range around the systolic peak (e.g. +/- 160ms), we expect no influence of the signal from the diastolic peak, and therefore the signal should be relatively symmetric and skewness relatively low. If the skewness of the systolic peak exceeds this threshold, we filter the peak from the list because we assume that it is a false peak, most likely a diastolic peak misclassified as a systolic peak.

Next, we consider an entire peak to peak interval. In this instance, both the systolic and diastolic peaks should be influencing the signal, causing a positive skewness (because the systolic peak should be larger than the diastolic peak). If the skewness for the RR interval is negative, the signal quality is low, and the RR interval is filtered from the list.

5. Filter RR intervals based on normalized RR speed

Despite the previous filtering steps, it is still possible for the diastolic peak to be misidentified as a systolic peak. To remove these artifacts, we deploy an algorithm similar to Lipponen and Tarvainen [54]–[56] that examines the RR series, which is a time series of successive RR intervals. We compare the current interbeat interval (IBI) to the 3 most recent IBIs. If the interbeat interval is within a 150ms threshold of any of the 3 most recent IBIs, it is considered valid. If not, then we attempt to combine the current IBI with the next IBI and recheck to see if the new combined IBI is within the 150ms threshold.

6. Calculate HR and PRV

With a full list of valid IBI intervals, it is trivial to compute the heart rate, and pulse rate variability. Our features use a sliding 12.5 second window to calculate both heart rate and pulse rate variability, where heart rate = 1/avg (IBI) and pulse rate variability can be characterized as the successive differences of normal beats (SDNN), and root mean square of successive differences (RMSSD) over the 12.5 second window. Note that we use the term pulse rate variability instead of heart rate variability since we are measuring heart activity through the circulatory system, which layers a variable pulse transit time (PTT) to the pulse rate variability statistic.

7. Calculate Respiratory Rate

To calculate respiratory rate, we take the trend signal from the seasonal decomposition in an earlier step, and then run peak detection using the Python scipy.signal.find_peaks algorithm, with a specified distance between peaks corresponding to a maximum of 50 breaths per minute, and a minimum peak prominence of the standard deviation / 5. The respiratory intervals (RI) are calculated as the successive differences between the peaks, and the respiratory rate = 60 / avg(RI) in breaths per minute.

Table 1. *List of features organized by sensor and feature family. Features in bold were included in models*

Sensor	Feature Family	Feature	Unit
Pupillometry	Pupil Size	**Pupil Diameter (mean)**	**mm**
		Pupil Diameter (std)	**mm**
Eye Tracking	Discrete Wavelet Transformation	Reverse Biorthogonal Wavelet 3.1	freq/time
		haar	freq/time
		Reverse Biorthogonal Wavelet 6.8	freq/time
		ipa	freq/time
	Blink	**Blink Rate**	**Hz**

		Longest Blink Duration	**Hz**
		Blink Depth	**sec**
	Saccade	Number of Saccades (mean)	num./time
		Number of Saccades (std)	num./time
		Path length	**degree**
		Duration (mean)	**sec**
		Duration (std)	sec
		Rate	**Hz**
		Length (longest)	sec
		Length (sum)	sec
		Saccade length (mean)	sec
		Speed (mean)	**sec**
		Speed (std)	**sec**
		Speed (max)	**sec**
	Fixation	Duration (longest)	sec
		Duration (sum)	sec
		Duration (mean)	sec
		Duration (std)	sec
	Fixation/Saccade Ratio	Rate	sec
		Duration	sec
Pulse Plethsmography	Heart Rate	**Interbeat Interval (IBI)**	**ms**
		Standard Deviation of Normal Beats (SDNN)	**ms**
	Heart Rate Variability	**Standard Deviation of Successive Differences (SDSD)**	**ms**
		Root Mean Square of Successive Differences (RMSSD)	**ms**
		Avg. Power Spectral Density [0.0-0.2] (mean)	**Hz**
		Avg. Power Spectral Density [0.2-0.4] (mean)	**Hz**
		Avg. Power Spectral Density [0.4-0.6] (mean)	**Hz**
		Avg. Power Spectral Density [0.6-0.8] (mean)	**Hz**
		Avg. Power Spectral Density [1-2] (mean)	**Hz**
	Respiration	Respiration Rate	num./time

3.3 Filtering.

1. Ordinal compliance

One of the advantages of our stimulus regime is that the difficulty of the task objectively increases; the easier tasks are entirely contained within the more difficult tasks. We expect subjective ratings of mental effort to follow suit: the easiest task should be rated as less demanding than the medium task which should be rated less demanding than the hard task. Using

this knowledge allowed us to evaluate the quality of our subjective ratings[2] and increased our confidence with using them in our labeling paradigm. For each participant, we defined the *ordinal compliance* of subjective ratings by evaluating whether the average subjective rating for each task (each task was repeated three times) adhered to the expected ordinal relationship. That is, for each individual, the easy task averages the lowest subjective rating, and the hardest task averages to the highest subjective rating. All of the data included in our test dataset is ordinally compliant with respect to subjective ratings.

2. Filtering for pupil signal quality

To minimize the use of noisy pupil data in our dataset, we devised a filtering mechanism based on the assumption that left and right eye pupil dilation should be highly correlated. First, a running average of the difference between *left_pupil_dilation* and *right_pupil_dilation* is calculated over a 120 sample (~1 second) rolling window. Next, the standard deviation of the window is calculated. If the standard deviation exceeds a threshold of 0.3, the window is considered too noisy. The percentage of noisy windows in a session is calculated by dividing the number of noisy windows across the total number of windows. Then if the percentage of noisy windows exceeds 30% of the overall data, the entire data associated with the subject is excluded from training. All of the data included in this dataset has passed this signal quality filter.

[2] Subjective reports, while considered the gold standard for understanding the momentary experiences of individuals, are rife with inconsistencies, mistakes, and often discarded as not "objective" enough to be used to estimate ground truth in machine learning research.

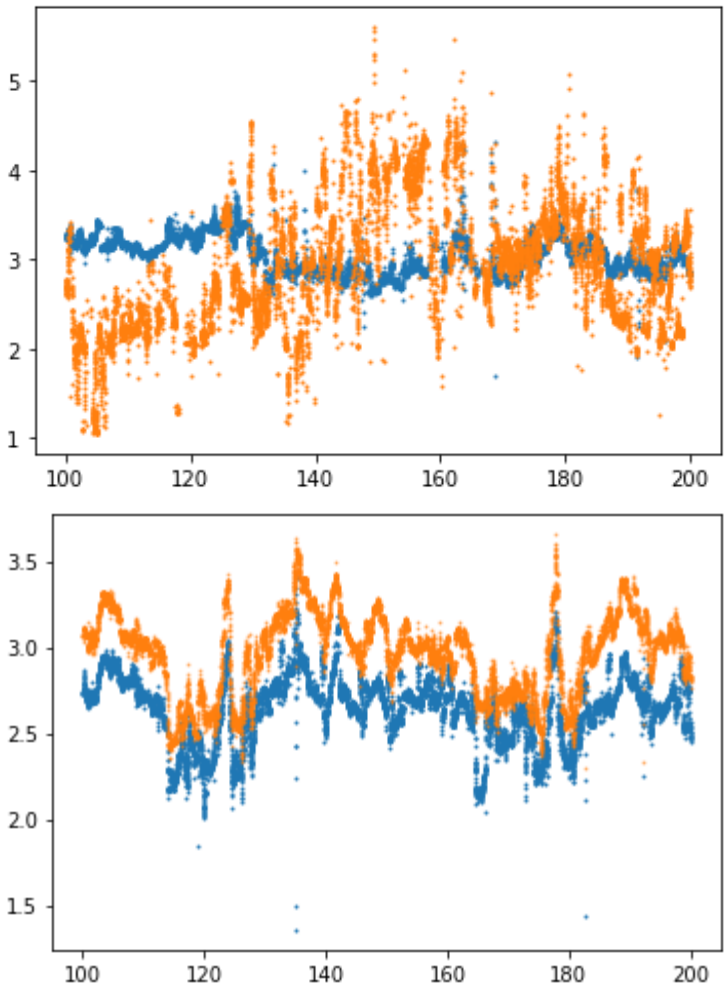

Figure 3. Example pupil signal quality for filtering (Top) Example of noisy pupil data that would be counted as noise for our pupil signal quality filtering. (Bottom) Example of clean pupil data that would pass the filtering. The blue lines represent the pupil diameter for the left eye, and the orange lines represent the pupil diameter for the right eye.

3.4 Labeling Cognitive Load.

One of the biggest challenges for predicting cognitive load in a context independent way is that it is very difficult to ascertain "ground truth" cognitive load levels. This is a challenge for data quality, especially label quality, because the classifier is trained on the labeled data. The closer the labels are to the "true" measure of interest, the more likely that the classifier will perform well on unlabeled data. We have implemented a multi-pronged labeling approach (e.g., [57]) that combines indices of task difficulty and subjective ratings of mental load to create a labeling paradigm that most reliably represents the true level of cognitive load experienced by individuals in a given moment. Once we validated the subjective ratings, we used them in combination with task difficulty to label cognitive load for each individual trial.

1. *Normalization of subjective ratings for each individual between [0,1].*

Using z-score normalization, the resulting data distribution is centered at 0 with 95% of the samples between [-2,2]. Our desired score is between [0,1], so we transform the z-score using the following formula.

$$\text{Normalized\_subjective\_rating} = (\text{z\_score\_rating} / 4) + 0.5$$

Although technically only 95% of the resulting samples are between the desired range of [0,1], this is sufficient for our needs.

2. Determine the relative difficulty of each task using population-wide statistics.

The tasks difficulty objectively increases from easy to hard. We assign numeric values to each task difficulty to space them equally within our regression space to determine the relative increase in difficulty between tasks.

Table 2. *Normalized difficulty rating for each level of task difficulty*

Task Description	Normalized difficulty rating
Task 1: Visual Vigilance	0.25
Task 2: Visual Vigilance + Arithmetic	0.50
Task 3: Visual Vigilance + Arithmetic + Audio Vigilance	0.75

3. Combine individual subjective rating with task averages using a weighted sum

Normalized Subjective Rating = population_wide_normalized_task_rating * 0.5 + individual_normalized_rating * 0.5

Figure 3 illustrates the resulting distribution of labels for the different cognitive load scores for each task difficulty condition. Because we only have one subjective rating for the entire task, we assume that the cognitive load score label is constant for the duration of the task.

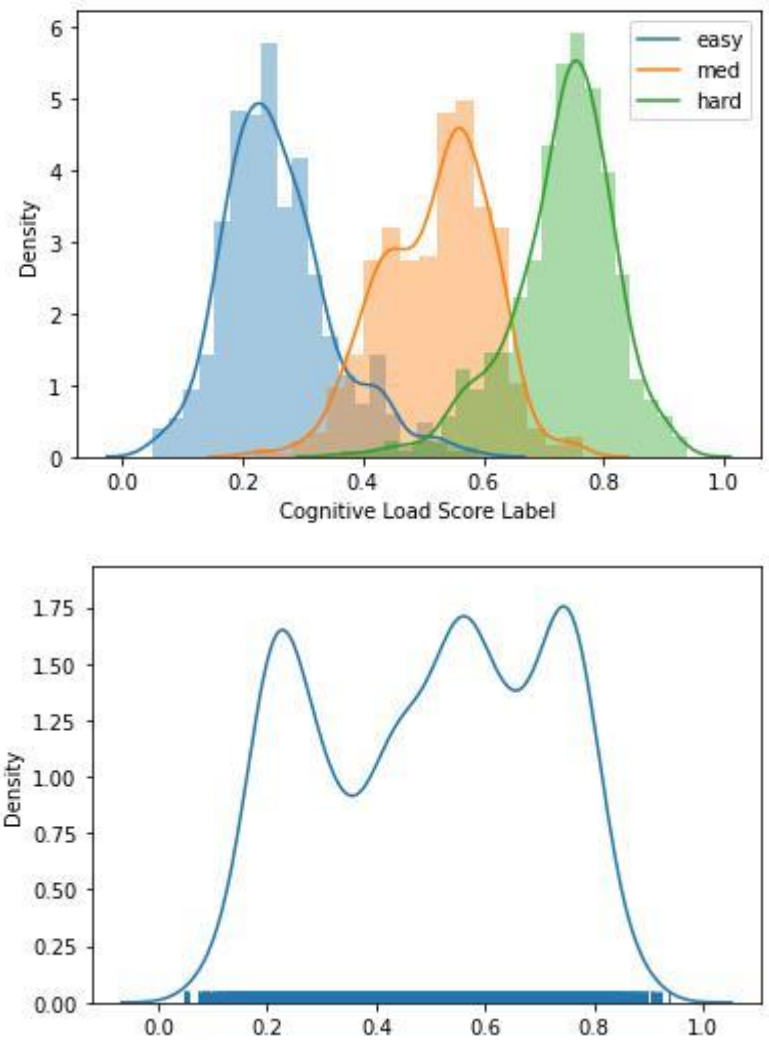

Figure 3. Density of labels separated by task difficulty.

4. Results

4.1 Data processing for machine learning model training. While the participant worked on the task, we collected eye tracking, pupillometry, and PPG recordings simultaneously. Figure 4 illustrates the corresponding signals and labels (task difficulty and normalized subjective rating) within one task. We use a sliding window, e.g. 12.5 second window and 1 second skip step, to create signal segments and associate corresponding labels with them. We will extract a set of features (using the pre-processing and extractions methods articulated in the previous section) from each signal segment and form it as a n-dimensional vector, where n represents the number of features. The task difficulty label is a discrete value with the options of "low", "medium", and "high", while normalized subjective rating is a continuous value with the range from 0 to 1.

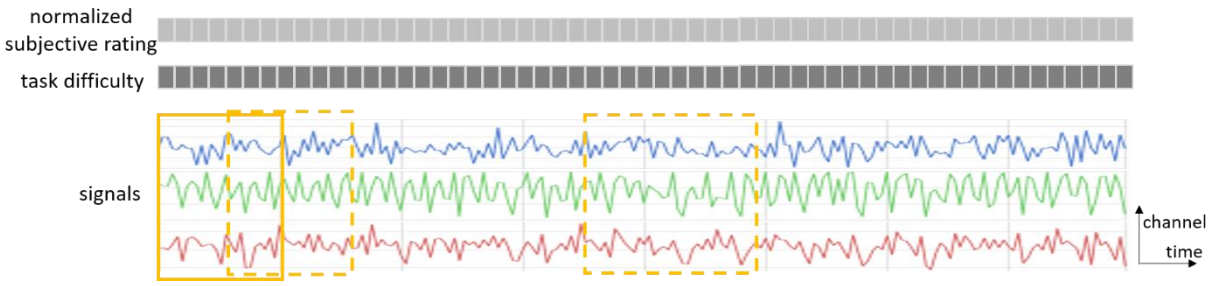

Figure 4. Signal samples and corresponding labels

4.2 Analysis of Features and Labels. As a first step, we investigated how well our subjective labels related to task difficulty levels. Figure 3 shows the distribution of normalized subjective ratings across tasks of "low", "medium" and "high" difficulties. As we can see, the distribution of labels clusters around three centers which correspond to the three task difficulty levels.

Next, we analyzed the relative importance of each feature in the overall model. These results are presented in Figure 7 which shows the results of a permutation feature importance technique to calculate the importance of individual features on unseen test or validation data. ([58], [59]Altman, et al, 2010; Huang, Lu, & Xu, 2016). Using this technique, we can learn the importance of individual features in the model by permuting the value of a single feature and noting the drop in accuracy in the overall model. Our results suggest that of the five most important features there are four eye tracking features (mean saccade speed, standard deviation of saccade duration, mean saccade duration) and one heart rate variability feature (power spectral density from 0.2-0.4). All of these high importance features are included in our final model along with additional features to provide more robustness in model performance (20 features in total: 11 eye tracking features, 9 PPG features, see Table 1 for a full list of features).

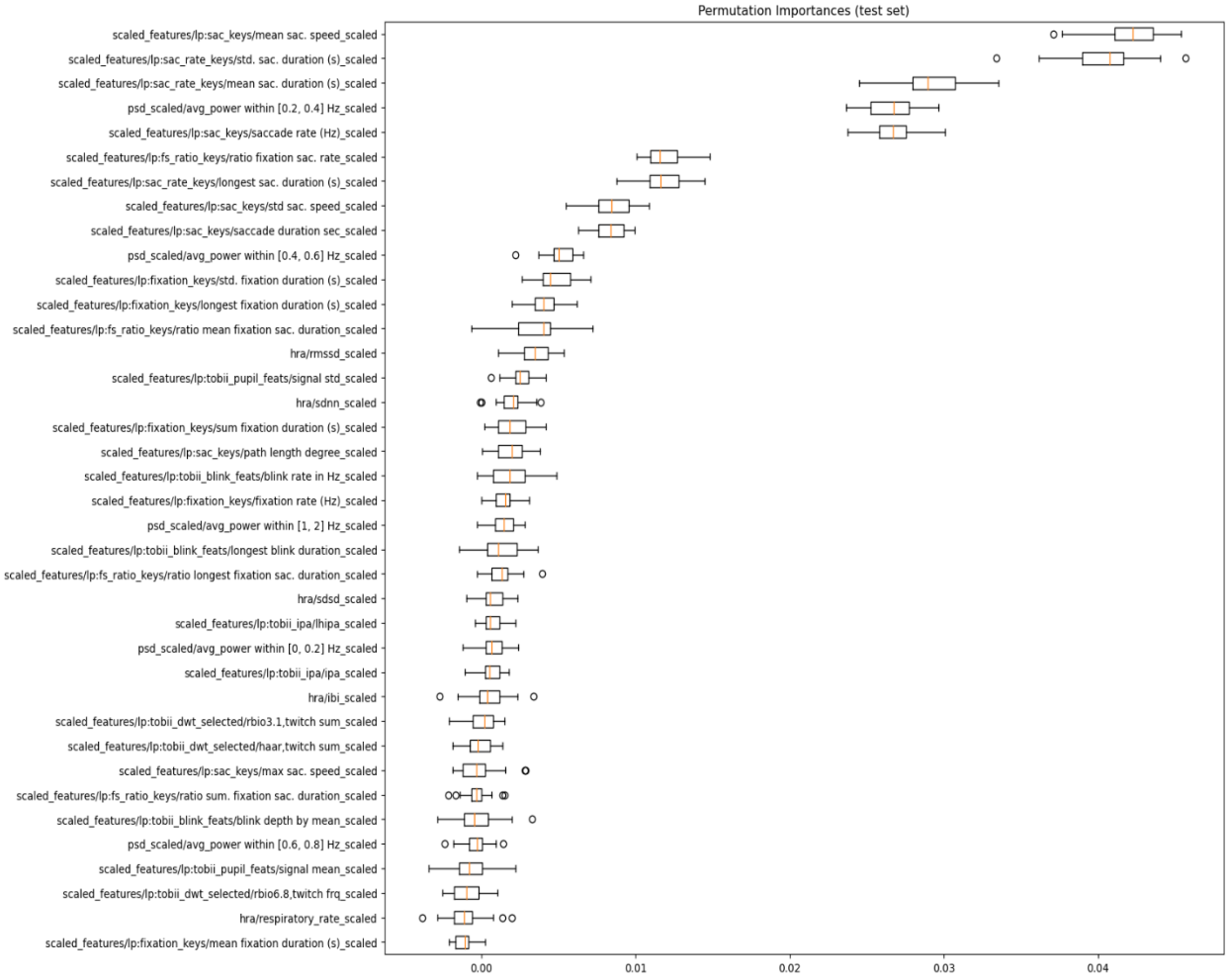

Figure 5. Results of a permutation feature importance technique to calculate the importance of individual features (listed in Table 1) on unseen test or validation data.

4.3 Machine learning model training. We trained a machine learning model with a sequence of n-dimensional features to predict the task difficulty level and the demanding cognitive load. The model also quantifies its uncertainty in each prediction. Figure 6 illustrates the machine learning model architecture.

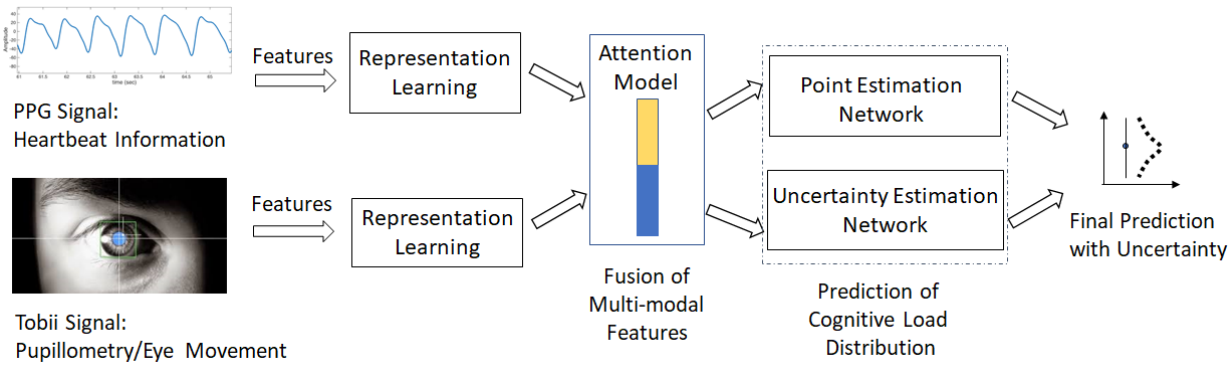

Figure 6. High-level representation of machine learning model architecture from signals to prediction

Within the architecture, the first step is to learn high-level representations of each signal modality from its input sequences in the form of t by n tensors, where t represents the number of sequence steps and n represents the number of features. We leverage 1-dimensional convolutional networks to extract local dependency patterns from the input sequences and turn the inputs to a multi-dimensional vector of m. Specifically, we use multiple blocks of 1D-CNN plus ReLu activation layers, after which a global pooling layer is applied. Secondly, once representations are learned for each input modality, these representations are fused together by leveraging an attention network, which is shown in Figure 7.

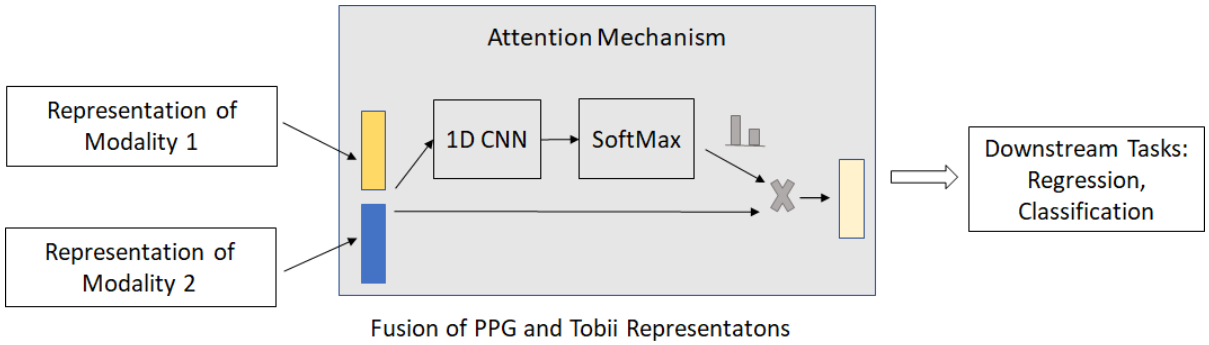

Figure 7. Illustration of high-level representation learning through sensor fusion

Third, in Figure 8, the fused representation is used to estimate a probabilistic distribution of the possible prediction values. We characterize this distribution as a Gaussian distribution, whose mean and standard deviation parameters can be modeled as two multiple layer perceptron neural networks. We use a gradient decent algorithm to train the machine learning model end to end with respect to maximizing the likelihood of normalized subjective report values.

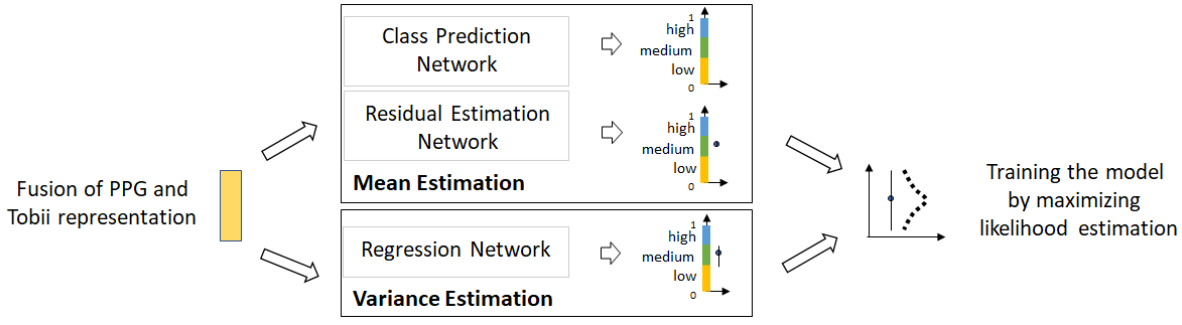

Figure 8. Depicts architecture from fused representation through cognitive load distribution prediction to final prediction with uncertainty

4.4 Machine learning model results. Table 4 shows the results of inference engine performance in terms of mean absolute error, accuracy, and uncertainty. Mean absolute error represents the average error of our inferences when predicting participant cognitive load value on a range from 0 to 1. Accuracy represents a fraction of the correct model predictions. Prediction uncertainty is quantified by the standard deviation of the gaussian distribution that our model is trying to predict. Currently, the inference engine trained on 11 Tobii features outperforms that trained 9 PPG features but the fused model still outperforms either individually (see Table 1 for list of features included in the model). As we can see, when we use both Tobii features and PPG features for inference engine training, the mean absolute error decreases while prediction accuracy increases. Meanwhile, the model prediction uncertainty goes down.

Table 4. Inference Engine Evaluation using Multimodal Inputs

	Model based on fusion of PPG and Tobii features	**Model based on Tobii features**	**Model based on PPG features**
Regression Error (Mean Absolute Error)	0.1106	0.1131	0.1669
Classification Model (Accuracy)	79.08%	73.45%	54.56%

Conclusion

In this technical report, we report the results of a large-scale experiment to develop a model that can reliably predict real time mental effort in VR. We collected physiological, self-report, and task data from more than seven hundred participants who completed mentally demanding tasks of increasing difficulty. Using an innovative filter and labeling pipeline, we developed a multi-modal, fusion model that predicted real time cognitive load with high accuracy and low uncertainty. As part of this report, we are releasing our "test dataset", that is the dataset that we used to test our trained model, which includes multimodal data from 100 users. With this release, we hope to spark interest and excitement in HP Omnicept but also to validate our process, pipeline, and results with the larger research and technical community.

We will continue to improve and update our inference engines over time to account for some of the limitations in this dataset. First, since this open dataset was collected, we have expanded our data collection activities to include data from several additional locations across the world, including data from participants in Africa, Asia, and North America. We are committed to representative sampling to mitigate pervasive bias in AI [48], [49] and one way to do that is by collecting data from as wide a swath of the general population as possible. Second, we will continue to expand the cognitive load testing contexts. Our current multi-tasking paradigm varies in both difficulty and modality but we have not, yet, tested our models in other types of cognitive load contexts (e.g., with time or social evaluative pressure, with motor load, with continuously increasing difficulty). We look forward to including new manipulations of load in future releases. Third, we are actively expanding this work to exploit the particular features of virtual environments. Specifically, we are teasing apart physiological responses in differing lighting conditions. The muscles of the pupil, for example, respond to both changes in

light and changes in sympathetic activation. VR affords the ability to model and separate the influence of each. In the meantime, we have found that removing pupil features from the model results in similar performance, making the model less sensitive to lighting changes.

We hope this technical report and dataset proves useful and fruitful for technologists, researchers, and machine learning scientists and that this engenders spirited discourse with the larger scientific community. We encourage researchers of all stripes to test, validate, and reach out to our team with questions, thoughts, or insights.

References

[1] J. Sweller and P. Chandler, "Evidence for cognitive load theory," *Cogn. Instr.*, vol. 8, no. 4, pp. 351–362, 1991.

[2] J. Sweller, "Cognitive load theory," in *Psychology of learning and motivation*, vol. 55, Elsevier, 2011, pp. 37–76.

[3] S. Chen, "A Comparison of Four Methods for Cognitive Load Measurement," pp. 76–79, 2011.

[4] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information.," *Psychol. Rev.*, vol. 63, no. 2, p. 81, 1956.

[5] R. M. Yerkes and J. D. Dodson, "The relation of strength of stimulus to rapidity of habit-formation," *Punishm. Issues Exp.*, pp. 27–41, 1908.

[6] K. H. Teigen, "Yerkes-Dodson: A Law for all Seasons," *Theory Psychol.*, vol. 4, no. 4, pp. 525–547, 1994, doi: 10.1177/0959354394044004.

[7] A. D. Baddeley and G. Hitch, "Working memory," in *Psychology of learning and motivation*, vol. 8, Elsevier, 1974, pp. 47–89.

[8] F. Paas, J. E. Tuovinen, H. Tabbers, and P. W. M. Van Gerven, "Cognitive load measurement as a means to advance cognitive load theory," *Educ. Psychol.*, vol. 38, no. 1, pp. 63–71, 2003.

[9] N. Cowan, "The Magical Mystery Four: How is Working Memory Capacity Limited, and Why?," doi: 10.1177/0963721409359277.

[10] R. W. Engle and M. J. Kane, "Executive attention, working memory capacity, and a two-factor theory of cognitive control," *Psychol. Learn. Motiv.*, vol. 44, pp. 145–200, 2004.

[11] K. Oberauer, H.-M. Süß, O. Wilhelm, and N. Sander, "Individual differences in working memory capacity and reasoning ability.," in *Variation in working memory.*, New York, NY, US: Oxford University Press, 2007, pp. 49–75.

[12] M. Iskander, "Burnout, Cognitive Overload, and Metacognition in Medicine," *Medical Science Educator*, vol. 29, no. 1. Springer, pp. 325–328, Mar. 15, 2019, doi: 10.1007/s40670-018-00654-5.

[13] J. Sweller, "Cognitive load theory, learning difficulty, and instructional design," *Learn. Instr.*, vol. 4, no. 4, pp. 295–312, 1994.

[14] P. Ayres, "Impact of reducing intrinsic cognitive load on learning in a mathematical domain," *Appl. Cogn. Psychol. Off. J. Soc. Appl. Res. Mem. Cogn.*, vol. 20, no. 3, pp. 287–298, 2006.

[15] R. McKendrick, B. Feest, A. Harwood, and B. Falcone, "Theories and Methods for Labeling Cognitive Workload: Classification and Transfer Learning," *Front. Hum. Neurosci.*, vol. 13, no. September, pp. 1–20, 2019, doi: 10.3389/fnhum.2019.00295.

[16] P. Hepsomali *et al.*, "Cognitive load measurement as a means to advance cognitive load theory," *Educ. Psychol.*, vol. 38, no. 1, pp. 63–71, Jul. 2003, doi: 10.1016/j.biopsycho.2017.05.013.

[17] S. Afzal and P. Robinson, “Natural affect data: Collection and annotation,” in *New perspectives on affect and learning technologies*, Springer, 2011, pp. 55–70.

[18] P. Biswas and G. Prabhakar, “Detecting drivers’ cognitive load from saccadic intrusion,” *Transp. Res. part F traffic Psychol. Behav.*, vol. 54, pp. 63–78, 2018.

[19] T. M. Sezgin and P. Robinson, “Affective video data collection using an automobile simulator,” in *International Conference on Affective Computing and Intelligent Interaction*, 2007, pp. 770–771.

[20] I. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang, “Facial expression recognition from video sequences: temporal and static modeling,” *Comput. Vis. image Underst.*, vol. 91, no. 1–2, pp. 160–187, 2003.

[21] L. F. Barrett, R. Adolphs, S. Marsella, A. M. Martinez, and S. D. Pollak, “Emotional expressions reconsidered: Challenges to inferring emotion from human facial movements,” *Psychol. Sci. public Interes.*, vol. 20, no. 1, pp. 1–68, 2019.

[22] H. Boril, S. O. Sadjadi, and J. H. L. Hansen, “UTDrive: Emotion and cognitive load classification for in-vehicle scenarios,” 2011.

[23] J. Su and S. Luz, “Predicting cognitive load levels from speech data,” in *Recent Advances in Nonlinear Speech Processing*, Springer, 2016, pp. 255–263.

[24] M. Gjoreski *et al.*, “Datasets for Cognitive Load Inference Using Wearable Sensors and Psychological Traits,” *Appl. Sci.*, vol. 10, no. 11, p. 3843, 2020.

[25] W. L. Romine *et al.*, “Using machine learning to train a wearable device for measuring students’ cognitive load during problem-solving activities based on electrodermal activity, body temperature, and heart rate: Development of a cognitive load tracker for both personal and cla,” *Sensors (Switzerland)*, vol. 20, no. 17, pp. 1–14, 2020, doi: 10.3390/s20174833.

[26] E. Haapalainen, S. Kim, J. F. Forlizzi, and A. K. Dey, “Psycho-physiological measures for assessing cognitive load,” in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, 2010, pp. 301–310.

[27] L. Fridman, B. Reimer, B. Mehler, and W. T. Freeman, “Cognitive load estimation in the wild,” *Conf. Hum. Factors Comput. Syst. - Proc.*, vol. 2018-April, no. 1, 2018, doi: 10.1145/3173574.3174226.

[28] P. Jerčić, C. Sennersten, and C. Lindley, “Modeling cognitive load and physiological arousal through pupil diameter and heart rate,” *Multimed. Tools Appl.*, vol. 79, no. 5–6, pp. 3145–3159, 2020, doi: 10.1007/s11042-018-6518-z.

[29] E. Ferreira *et al.*, “Assessing real-time cognitive load based on psycho-physiological measures for younger and older adults,” in *2014 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*, 2014, pp. 39–48.

[30] R. D. Dias, M. A. Zenati, R. Stevens, J. M. Gabany, and S. J. Yule, “Physiological synchronization and entropy as measures of team cognitive load,” *J. Biomed. Inform.*, vol. 96, p. 103250, 2019.

[31] J. A. Healey and R. W. Picard, “Detecting stress during real-world driving tasks using physiological sensors,” *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 156–166, 2005.

[32] C. Setz, B. Arnrich, J. Schumm, R. La Marca, G. Tröster, and U. Ehlert, "Discriminating stress from cognitive load using a wearable EDA device," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 2, pp. 410–417, 2009.

[33] K. P. Madore *et al.*, "Memory failure predicted by attention lapsing and media multitasking," *Nature*, no. January, 2020, doi: 10.1038/s41586-020-2870-z.

[34] N. Nourbakhsh, Y. Wang, and F. Chen, "GSR and Blink Features," pp. 159–166, 2013.

[35] A. Jimenez-Molina, C. Retamal, and H. Lira, "Using psychophysiological sensors to assess mental workload during web browsing," *Sensors*, vol. 18, no. 2, p. 458, 2018.

[36] M. U. Ahmed, S. Begum, R. Gestlöf, H. Rahman, and J. Sörman, "Machine Learning for Cognitive Load Classification–A Case Study on Contact-Free Approach," in *IFIP International Conference on Artificial Intelligence Applications and Innovations*, 2020, pp. 31–42.

[37] K. Pettersson, J. Tervonen, J. Närväinen, P. Henttonen, I. Määttänen, and J. Mäntyjärvi, "Selecting Feature Sets and Comparing Classification Methods for Cognitive State Estimation," in *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2020, pp. 683–690, doi: 10.1109/BIBE50027.2020.00115.

[38] B. C. O. F. Fehringer, "One threshold to rule them all? Modification of the Index of Pupillary Activity to optimize the indication of cognitive load," in *ACM Symposium on Eye Tracking Research and Applications*, 2020, pp. 1–5.

[39] A. T. Duchowski *et al.*, "The Index of Pupillary Activity: Measuring Cognitive Load *Vis-à-Vis* Task Difficulty with Pupil Oscillation," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018, pp. 1–13, doi: 10.1145/3173574.3173856.

[40] E. Stuyven, K. Der Van Goten, A. Vandierendonck, K. Claeys, and L. Crevits, "The effect of cognitive load on saccadic eye movements," *Acta Psychol. (Amst).*, vol. 104, no. 1, pp. 69–85, Mar. 2000, doi: 10.1016/S0001-6918(99)00054-2.

[41] J. Zagermann, U. Pfeil, and H. Reiterer, "Measuring cognitive load using eye tracking technology in visual computing," in *Proceedings of the sixth workshop on beyond time and errors on novel evaluation methods for visualization*, 2016, pp. 78–85.

[42] C. Ranti, W. Jones, A. Klin, and S. Shultz, "Blink Rate Patterns Provide a Reliable Measure of Individual Engagement with Scene Content," pp. 1–10, 2020, doi: 10.1038/s41598-020-64999-x.

[43] M. R. Islam, S. Barua, M. U. Ahmed, S. Begum, and G. Di Flumeri, "Deep learning for automatic EEG feature extraction: an application in drivers' mental workload classification," in *International Symposium on Human Mental Workload: Models and Applications*, 2019, pp. 121–135.

[44] M. R. Islam *et al.*, "A Novel Mutual Information Based Feature Set for Drivers' Mental Workload Evaluation Using Machine Learning," *Brain Sci.*, vol. 10, no. 8, p. 551, 2020.

[45] L. Zhang *et al.*, "Cognitive load measurement in a virtual reality-based driving system for autism intervention," *IEEE Trans. Affect. Comput.*, vol. 8, no. 2, pp. 176–189, 2017.

[46] P. Sarkar, K. Ross, A. J. Ruberto, D. Rodenbura, P. Hungler, and A. Etemad,

"Classification of cognitive load and expertise for adaptive simulation using deep multitask learning," in *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2019, pp. 1–7.

[47] A. Saha, V. Minz, S. Bonela, S. R. Sreeja, R. Chowdhury, and D. Samanta, "Classification of eeg signals for cognitive load estimation using deep learning architectures," in *International Conference on Intelligent Human Computer Interaction*, 2018, pp. 59–68.

[48] S. Ransbotham, "The Subtle Sources of Sampling Bias Hiding in Your Data," *MIT Sloan Manag. Rev.*, vol. 59, no. 1, 2017.

[49] D. Roselli, J. Matthews, and N. Talagala, "Managing bias in AI," in *Companion Proceedings of The 2019 World Wide Web Conference*, 2019, pp. 539–544.

[50] M. Bartels and S. P. Marshall, "Measuring Cognitive Workload across Different Eye Tracking Hardware Platforms," in *Proceedings of the Symposium on Eye Tracking Research and Applications*, 2012, pp. 161–164, doi: 10.1145/2168556.2168582.

[51] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," in *Advances in psychology*, vol. 52, Elsevier, 1988, pp. 139–183.

[52] S. Mathôt, E. Aarts, M. Verhage, J. V. Veenvliet, C. V. Dolan, and S. van der Sluis, "A simple way to reconstruct pupil size during eye blinks," *Figshare*, vol. 17, no. April 2013, pp. 491–496, 2013, [Online]. Available: http://www.nature.com/neuro/journal/v17/n4/full/nn.3648.html%5Cnhttp://www.nature.com/neuro/journal/v17/n4/pdf/nn.3648.pdf%5Cnhttp://www.nature.com/doifinder/10.1038/nn.3648.

[53] M. Elgendi, "Optimal signal quality index for photoplethysmogram signals," *Bioengineering*, vol. 3, no. 4, pp. 1–15, 2016, doi: 10.3390/bioengineering3040021.

[54] J. A. Lipponen and M. P. Tarvainen, "A robust algorithm for heart rate variability time series artefact correction using novel beat classification," *J. Med. Eng. Technol.*, vol. 43, no. 3, pp. 173–181, 2019.

[55] M. P. Tarvainen, J.-P. Niskanen, J. A. Lipponen, P. O. Ranta-Aho, and P. A. Karjalainen, "Kubios HRV—a software for advanced heart rate variability analysis," in *4th European conference of the international federation for medical and biological engineering*, 2009, pp. 1022–1025.

[56] M. P. Tarvainen, J.-P. Niskanen, J. A. Lipponen, P. O. Ranta-Aho, and P. A. Karjalainen, "Kubios HRV–heart rate variability analysis software," *Comput. Methods Programs Biomed.*, vol. 113, no. 1, pp. 210–220, 2014.

[57] J. N. Bailenson, K. Swinth, C. Hoyt, S. Persky, A. Dimov, and J. Blascovich, "The independent and interactive effects of embodied-agent appearance and behavior on self-report, cognitive, and behavioral markers of copresence in immersive virtual environments," *Presence Teleoperators Virtual Environ.*, vol. 14, no. 4, pp. 379–393, 2005.

[58] A. Altmann, L. Toloşi, O. Sander, and T. Lengauer, "Permutation importance: a corrected feature importance measure," *Bioinformatics*, vol. 26, no. 10, pp. 1340–1347, 2010.

[59] N. Huang, G. Lu, and D. Xu, "A permutation importance-based feature selection method

for short-term electricity load forecasting using random forest," *Energies*, vol. 9, no. 10, p. 767, 2016.