D3.7 – SDK for integration of affect recognition into VR experiences

Work Package 3 – Lighthouse Projects at Laboratory Nodes

Authors:

Christof Lutteroth (UB)
Christopher Clarke (UB)
Crescent Jicol (UB)
Adwait Sharma (UB)
Dominic Potts (UB)

Grant Agreement number	**101070533**
Action Acronym	**EMIL**
Action Title	**European Media and Immersion Lab**
Call	**HORIZON-CL4-2021-HUMAN-01**

Funded by the European Union

UK Research and Innovation

EMIL project is funded by the European Union and UK Research and innovation.

Version date of the Annex I against which the assessment will be made	*Start date of the project*	*Due date of the deliverable*	*Actual date of submission*	*Lead BEN / AP for the deliverable*	*Dissemination level of the deliverable*
18.3.2022	1.9.2022	31.08.2023	31.08.2023	UB	Public

Document reviewer(s)	
Name	*Beneficiary*
Juhani Tenhunen	Aalto
Yu Xiao	Aalto

Abstract

This document, "*EmoSense* SDK for integration of **affect recognition** into VR experiences" (D3.7) provides an overview of a Software Development Kit (SDK) that can be used to collect physiological measures in real-time and produce a confidence value of predicted core affect for categorical and dimensional emotion models. The supported physiological measures include: pupillometry, gaze and head movement, heart rate and heart rate variability, galvanic skin response, and facial/lip gestures. The SDK includes software to take baseline and calibration measures for all physiological signals for the purpose of data cleaning and normalisation between users. This document will describe the software components of the SDK that support affect recognition in VR and give brief examples of integrating affect recognition into VR experiences.

1 Introduction

This document, "*EmoSense SDK for integration of affect recognition into VR experiences*" (D3.7) provides an overview of a Software Development Kit (SDK) used to integrate affect recognition into VR experiences based on several physiological metrics such as pupillometry, gaze and head movement, heart rate and heart rate variability, galvanic skin response, and facial/lip gestures. The SDK enables adaptive stimuli and Virtual Environments based on a User's predicted affective state. More specifically, confidence values for both categorical emotion models (Fear, Boredom, Happiness, Sadness, Stress, Excitedness, and Contentedness) and dimensional models (Pleasure-Arousal Scale[1]) are provided by the SDK based on Russel's Circumplex Model of Affect[2].

Herein, the document describes the software components of the SDK and supported hardware that provides the basis of our affect recognition model. We also provide brief examples of how the affect recognition SDK can be integrated into bespoke VR development projects or run as a background application to measure affect response in commercial VR experiences such as VR games. The document is divided into the following main chapters:

SDK Overview: An overview of the components that make up the SDK, the supported hardware and developer platforms, and external software dependencies.

Retrieving Real-Time Physiological Signals: Outlines how data is retrieved and processed from Bluetooth physiological sensors and how the SDK can be extended to support additional physiological sensors, and other Bluetooth Low Energy (BLE) devices such as IMUs and exercise machines.

Taking Baseline Physiological and Affect Measures: An overview of the calibration applications provided in the SDK for taking baseline measurements of various physiological metrics including pupil dilation, heart rate, and skin conductance.

Normalising data for interpersonal and environmental factors: A description of how physiological metrics are normalised to individual users to remove artefacts induced by interpersonal physiological differences and virtual environmental artefacts (such as pupil dilation caused by scene brightness).

[1] Bradley, M. M., & Lang, P. J. (1994). Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, *25*(1), 49-59.
[2] Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, *39*(6), 1161.

Physiological Post-Processing for Conventional VR Experiences: Summary of how the affect model can be integrated into a background VR application to measure user affect response in commercial VR experiences post hoc.

Application Examples: Examples of how the SDK can be integrated into Unity applications to design adaptive VR experiences built in Unity or measure affect response in existing commercial VR experiences.

2 *EmoSense* SDK Overview

The SDK is being maintained primarily for Unity, one of the most popular development platforms for VR experiences. The game engine uses scene trees, game objects, and components to manage assets and program behaviour in a VR scene. Different parts of the SDK are entirely modular that can be inserted into the scene as game objects to provide functionality for different parts of affect recognition. Image 1 provides an overview of the *EmoSense* SDK.

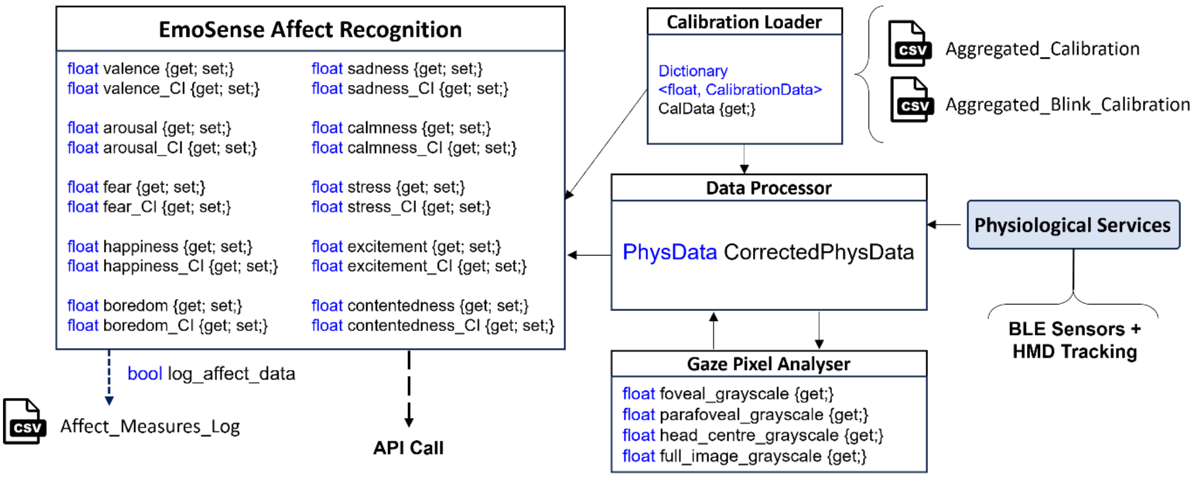

Image 1 An overview of the EmoSense SDK for real-time Affect Recognition

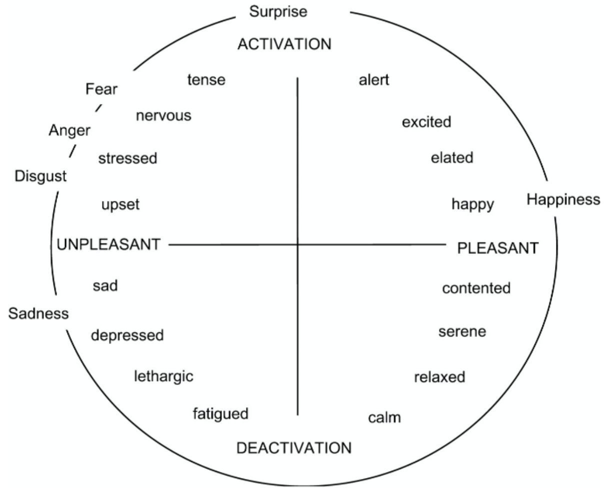

Image 2 Russell's Cicumplex Model of Affect which is the basis for the EmoSense Affect Recognition SDK

The boxes represent different components, with the top box denoting the component type and the bottom box detailing some of the variable information that can be retrieved by other components. Arrows represent function calls between different components of the SDK and blue boxes represent many components, such as *physiological services*.

The *EmoSense* Affect Recognition component is where API calls are made, and developers can retrieve predicted values and confidence intervals for core affect dimensional measures – valence and arousal – and categorical emotions based on Russell's circumplex model of affect (Image 2)[3].

As shown in Image 1, the *EmoSense* SDK comprises of the following additional components:

- **Physiological Services** – Sensor management and data retrieval module *(Section 3)*
- **Calibration Loader –** Loads user baseline physiological and affect measures *(Section 5)*
- **Data Processor –** Data filtering and normalisation module *(Section 5)*
- **Post-Processor Module –** Replaces the Physiological Services module to measure affect post hoc *(Section 6)*

Additionally, to normalise physiological signals and filter out interpersonal and environmental factors, a set of baseline measures must be taken. Within the *EmoSense* SDK, two executable Unity

[3] Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, *39*(6), 1161.

applications are provided for measuring baseline physiological. Both applications run directly on the VR HMD and produce calibration files specific to that user:

- A **blink calibration** program to take measures of the User's normal blink and gaze behaviour in the VR headset.
- A **pupil dilation** and **resting physiological metric calibration** program to take measures of pupil dilation under different lighting conditions and record a User's resting physiological measures.

The two calibration programs are further described in **Section 4**.

2.1 Supported Hardware and Platforms

The *EmoSense* SDK requires several different types of physiological sensors. Currently the SDK supports specific hardware but can be extended to include any generic BLE sensor and OpenVR supported HMD with eye tracking and pupillometry. Examples given in this document use the following hardware for affect recognition:

- **Polar H10** heart rate strap – provides BPM and RR (inter-beat) interval through BLE.
- **Shimmer3 GSR+** wearable galvanic skin response sensor – provides skin conductance and resistance through BLE.
- **HTC Vive Pro Eye** Virtual Reality Head-mounted Display – provides head and gaze position and direction, pupil dilation, and blink information.
- **HTC Vive Facial Tracker** mounted to the **Vive Pro Eye Headset** – provides lip pose information.

Additional peripherals supported by the SDK:

- **Wahoo KICKR Indoor Smart Bike Trainer** – not necessary for core affect recognition, but exercise machine information can be used by the SDK in conjunction with physiological data when engaging in VR Exergaming. Can be expanded to other BLE exercise machines.

EmoSense can be used with a variety of VR hardware including head and handsets supported by OpenVR (e.g. HTC Vive) and OpenXR (e.g. Oculus Quest). However, while the SDK can be used without eye tracking and pupillometry, we strongly recommend using the SDK with headsets that support this for more accurate affect prediction (e.g., HTC Vive Pro Eye, HP Reverb G2 Omnicept, or a VR headset with a third-party eye tracker such as Pupil Labs). The examples given in this document use the Unity game engine, but we plan to extend support to other game engines that use a similar Game Object and Component system such as Godot.

2.2 External Software Dependencies

The *EmoSense* Affect Recognition SDK relies on the following external software components for Bluetooth device connection and eye tracking:

- **Vive Sranipal** Eye and Lip Framework – Eye and Lip tracking in the VR headset
- **TobiiXR** SDK – additional Eye Tracking measures in the VR headset
- **BLEWinrtDll** – For BLE device connection to Unity
- **Shimmer API** – For Shimmer GSR device connection.

The subsequent sections of this documentation delve into the specifics of utilising each module of *EmoSense* and integrating affect recognition into various VR applications.

3 Retrieving Real-Time Physiological Signals

Image 4 shows a class diagram of the physiological sensing services provided by the *EmoSense* SDK and the data produced by each service. All device connections are maintained by the *Connection Manager* which also provides information of the connection status of each BLE sensor device and the eye and lip tracking system of the headset. All physiological services are part of a prefab game object in the *EmoSense* SDK that can be dragged into a Unity scene, the hierarchy for which is shown in Image 3.

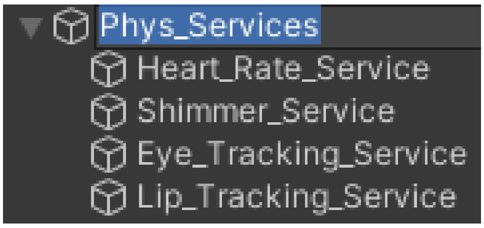

Image 3 Physiological Services Unity Prefab

Physiological Data across all sensors are sampled in Unity at the same rate as the application frame rate, which is approximately 90hz for most applications using the Vive Pro Eye headset. For most BLE devices, the sample rate is higher than 90hz, but for some sensors, such as the polar HR strap, the sample rate is much lower (1-2hz). Sampling is currently synchronised through a timer in Unity, at the time resolution of the current frame rate.

Each physiological service component is entirely modular, so less services can be used in the affect recognition model if the developer desires. Likewise, more can be added to expand the *EmoSense* SDK for example EEG or, in the context of exergaming, exercise machine data.

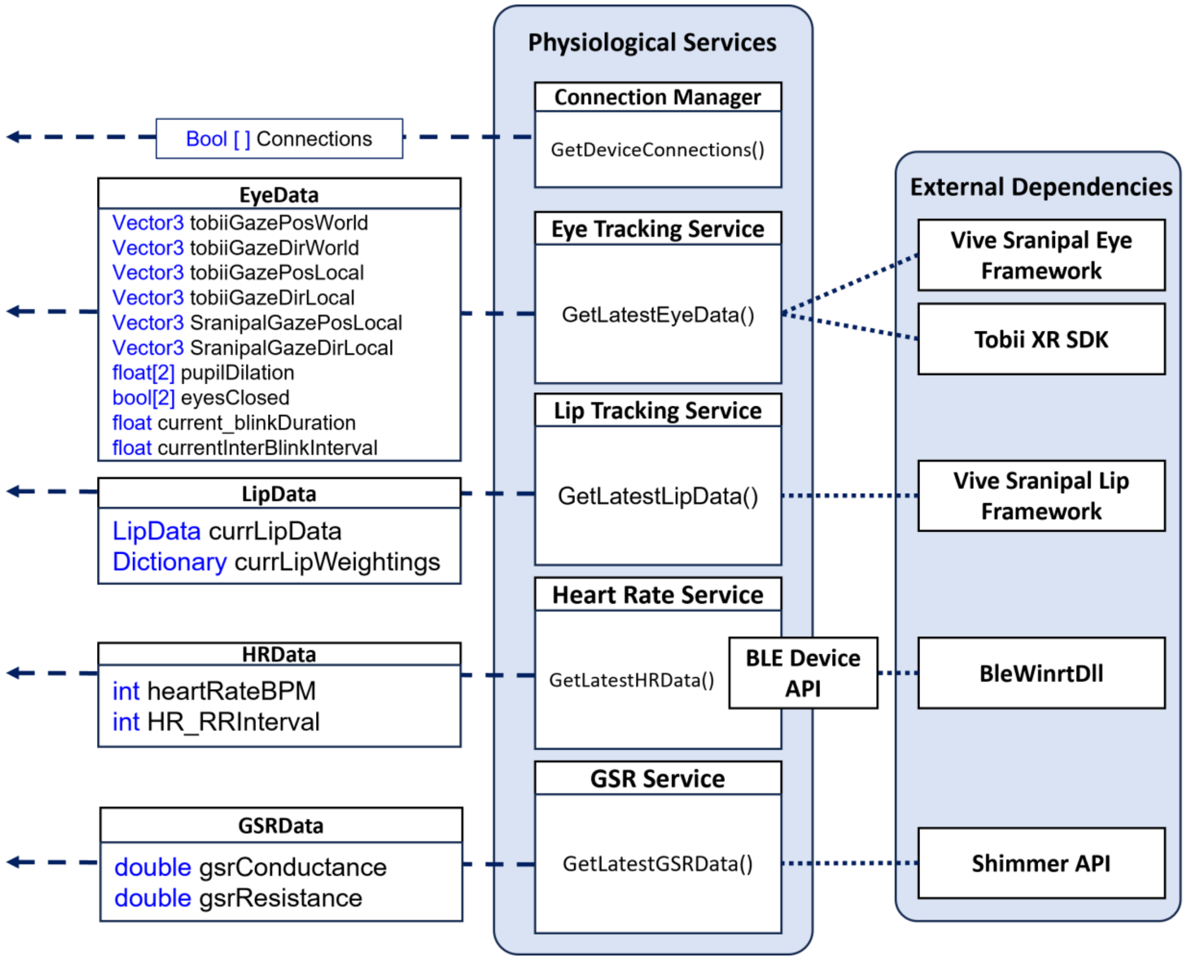

Image 4 Overview of the Physiological Service components and the available data in the EmoSense SDK that provide the basis of core affect prediction.

3.1 Eye Tracking Service

The Eye Tacking Service component uses the Vive Sranipal Eye framework and the Tobii XR SDK to retrieve information about the User’s eyes in real time. The EyeData object stores the latest information about the eyes including the gaze point in both local and world space as a Vector3, the current dilation in mm for both eyes as a floating point, information about the current blink rate (last blink duration and the current elapsed time of the inter-blink interval), and whether the eyes are closed. In principle, the *EmoSense* SDK only depends on Sranipal Eye Framework, but we included the Tobii XR SDK support to allow developers to use other VR headsets with Tobii eye tracking.

3.2 Lip Tracking Service

The Lip Tracking Service component uses the Vive Sranipal Lip Framework to retrieve information about different lip poses and gestures[4]. There are 26 different gestures that can be detected and the LipData object provides weighting as floating points values for each type of gesture. The result of the weightings can be rigged to a user's avatar in real-time. Primarily this is used to show lip movement on a virtual avatar while a user speaks, but this information can also be used to detect facial expressions that are induced by emotion such as surprise, disgust, or pleasure – which is incorporated into the affect recognition model.

3.3 Heart Rate Service

The Heart Rate Service component uses a BLE dynamic link library file which can be dropped into a Unity project. Expanding this part of the SDK to other BLE devices is just a case of changing the device name, service UUID and characteristic UUIDs to match the specific BLE device. For the specific heart sensor we use (Polar H10) we are able to retrieve information about the current beats-per-minute and the current RR (inter-beat) interval. These values allow the *EmoSense* SDK to calculate various HR variability values such as SDNN and RMSSD[5].

3.4 Galvanic Skin Response Service

Finally, the GSR Service component uses the Shimmer API to retrieve information on Galvanic Skin Response such as skin resistance, measured in kOhms, and skin conductance, measured in micro-Siemens, both represented as floating point values. The limitation of this is that the GSR service is not compatible with other types of GSR sensors, however BLE GSR devices can be incorporated by using the same API used by the Heart Rate Service.

4 Taking Baseline Physiological and Affect Measures

The *EmoSense* SDK provides the following two Unity applications designed to run on the Vive Pro Eye Head-Mounted Display that take physiological and affect baseline measures of a user and produces calibration CSV files that are designed to be used as input for the real-time affect recognition model.

---

[4] https://developer.vive.com/resources/vive-sense/eye-and-facial-tracking-sdk/
[5] Shaffer, F., & Ginsberg, J. P. (2017). An Overview of Heart Rate Variability Metrics and Norms. Frontiers in public health, 5, 258. https://doi.org/10.3389/fpubh.2017.00258

4.1 Blink Behaviour and Affect Baseline Measures

The blink calibration program, shown in Image 5, allows a user to take baseline measures of their natural blinking behaviour while using the VR HMD, primarily blink duration and the inter-blink intervals. The program first allows the User to calibrate the VR eye tracker through an HMD fitting check, inter-pupillary distance check, and a standard 5-point calibration provided by the Sranipal API. Then the user is placed in a neutral VR scene and administered a series of affect and emotion sampling questions for both categorical and dimensional models presented in the VR scene both visually and auditory, shown in Image 6. All affect questions are directly mapped to Russell's circumplex model (Image 2), with the categorical questions based on experience sampling literature[6] and the dimensional measures based on the widely used affect slider[7].

After sampling the users baseline affective state a simple visual stimulus, in this case an animated cube, is shown in front of the user which they look at for the duration of the calibration window (we recommend at least 2 minutes[8]). Once the calibration finishes, two calibration csv files are produced by the program: an aggregated file designed to be used as input for the affect recognition model with baseline blink and affect measures, and a raw file with data from every frame of the calibration.

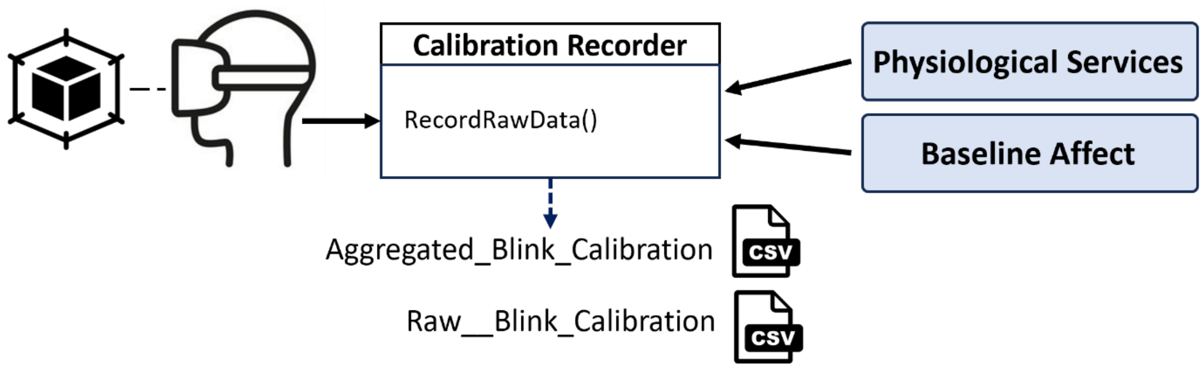

Image 5 An overview of the blink calibration procedure

[6] Csikszentmihalyi, M., Csikszentmihalyi, M., & Larson, R. (2014). Validity and reliability of the experience-sampling method. *Flow and the foundations of positive psychology: The collected works of Mihaly Csikszentmihalyi*, 35-54.
[7] Betella, A., & Verschure, P. F. (2016). The affective slider: A digital self-assessment scale for the measurement of human emotions. *PloS one*, *11*(2), e0148037.
[8] Cho, P., Sheng, C., Chan, C., Lee, R., & Tam, J. (2000). Baseline blink rates and the effect of visual task difficulty and position of gaze. *Current Eye Research*, *20*(1), 64-70.

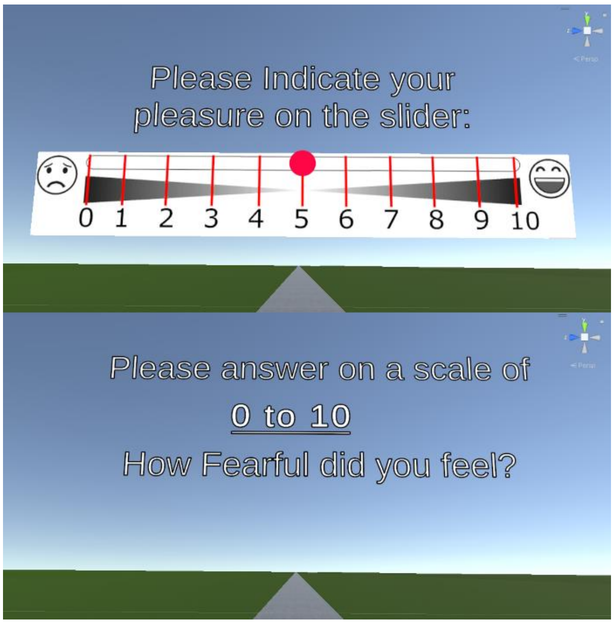

Image 6 Baseline affect sampling inside blink calibration program - affect slider and experience sampling questions

4.2 Pupil Dilation and Physiological Baseline Measures

The pupil dilation program (Image 6) is an additional calibration procedure which measures the users pupil dilation under different lighting conditions and takes baseline measures of the various physiological sensors while the user is at rest. The user first connects the devices to the corresponding physiological services and inputs their age. Optionally, the eye tracker can be recalibrated here in the same manner as the blink calibration program. Then the user runs the dilation calibration procedure in which they are shown 16 blank screens in increasing grayscale brightness (0 – 255 increasing in increments of 16). Each bright screen exposure is 2-3 seconds and the pupil dilation for that brightness level is sampled for 1 second of that window allowing the pupil time to adjust to the light which can be up to 1.5 seconds[9] [10]. Between each bright screen, the user is shown a black screen for 8 seconds to reset the dilation of the pupil.

[9] Raiturkar, P., Kleinsmith, A., Keil, A., Banerjee, A., & Jain, E. (2016, July). Decoupling light reflex from pupillary dilation to measure emotional arousal in videos. In *Proceedings of the ACM Symposium on Applied Perception* (pp. 89-96).
[10] Wang, C. A., & Munoz, D. P. (2021). Differentiating global luminance, arousal and cognitive signals on pupil size and microsaccades. *European Journal of Neuroscience*, *54*(10), 7560-7574.

While sampling the pupils, the program also takes measures of resting heart rate, galvanic skin response, and resting lip pose. The whole procedure takes approximately 4 minutes and produces two calibration csv files. The aggregated data file is used as input to the affect recognition model and contains median, mean, and standard deviation values for each physiological metric: dilation values under different lighting conditions, resting heart bpm, resting inter-beat interval, resting lip weightings for the 26 detected lip gestures, skin resistance and conductance, and user heart rate max and heart rate reserve. The raw data file contains frame-by-frame information for all sensor data.

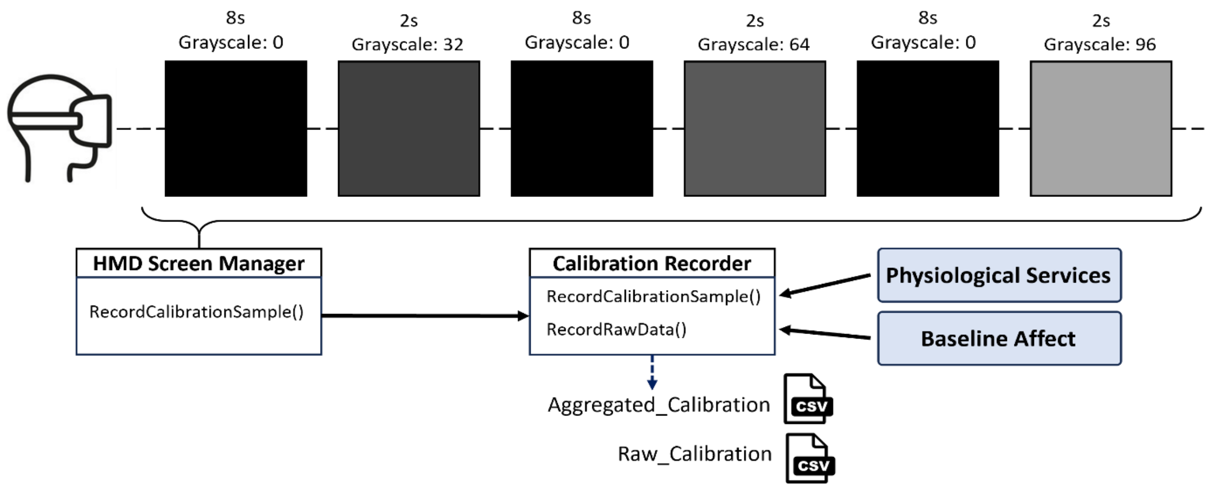

Image 7 An overview of the pupil dilation calibration procedure

5 Normalising data for interpersonal and environmental factors

To mitigate interpersonal and environmental factors, we take several data cleaning measures for affect prediction to improve correlation between physiological measures and emotions and avoid false negatives and false positives. Firstly, the affect and physiological baseline measures are a key input into the *EmoSense* affect recognition model and are compared to the real-time sensor data, the residual of which we assume is related to physical exertion and emotional response.

In addition to these baseline measures, the *DataProcessor* component in the SDK handles the filtering and normalizing of the data. Each physiological measure has to be filtered differently, for example the model takes the z-transform of both the pupil dilation and GSR conductance/resistance - due to large interpersonal differences in physiology. Likewise, we use a user's heart rate reserve to understand their heart rate range. We also apply two widely used filtering techniques, absolute and relative 'age-based' filtering, to a user's measured inter-beat (RR) interval, the filtering techniques for which are described by Karlsson et al. [11].

For pupil dilation, the *EmoSense* SDK has a *GazePixelAnalyser* component that samples the current brightness of the scene based on the User's gaze and head position. Specifically, it samples the grayscale brightness of pixels from several different positions relative to the user:

- User's foveal field of view based on gaze position - 2-degree visual angle.
- User's parafoveal field of view based on gaze position - 10-degree visual angle.
- User's parafoveal field of view based on head position (this assumes Users want to maintain a comfortable eye-in-head position) – 10-degree visual angle.
- User's entire field of view – entire camera image is sampled.

The gaze pixel analyser then looks up the corresponding baseline dilation value for both eyes based on the current sampled brightness for each type of field of view and compares the current dilation, see Image 8 for an overview. The residual dilation is assumed to be related to other factors aside from environmental brightness such as affective response.

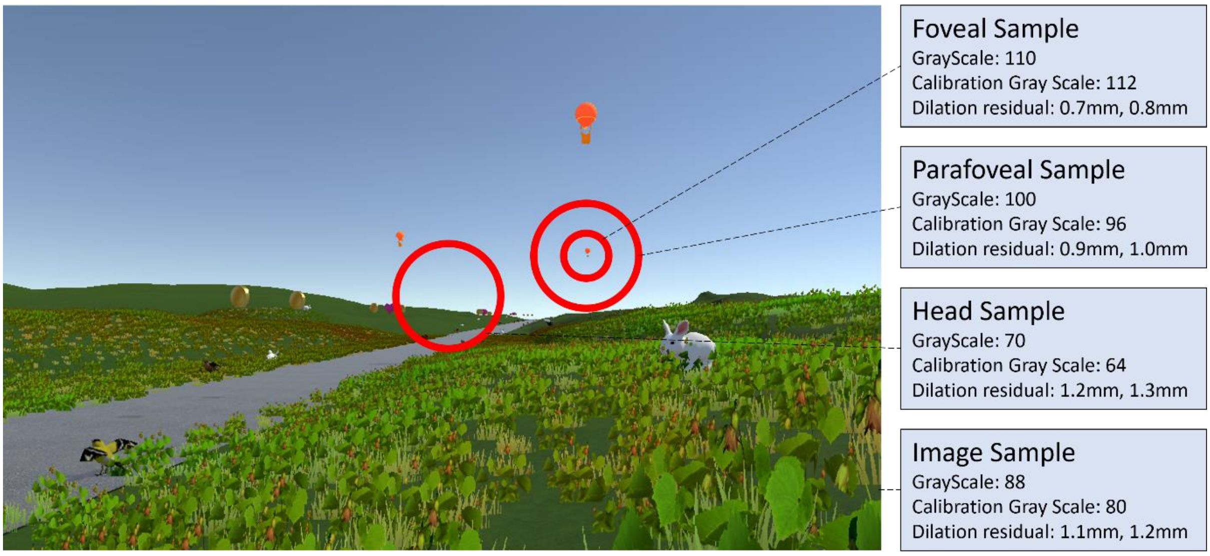

Image 8 Overview of different observed brightness values based on gaze and head position.

[11] Karlsson, M., Hörnsten, R., Rydberg, A., & Wiklund, U. (2012). Automatic filtering of outliers in RR intervals before analysis of heart rate variability in Holter recordings: a comparison with carefully edited data. *Biomedical engineering online*, *11*, 2. https://doi.org/10.1186/1475-925X-11-2

6 Physiological Post-Processing for Conventional VR Experiences

The final part of the *EmoSense* SDK is the *PostProcessing* module, that is designed to replace the physiological services component to allow for post hoc analysis of affect during a VR experience. This component is particularly useful for measuring emotions in conventional VR experiences, such as VR games available through Steam, where access to the headset view is not possible. SteamVR does not support multiple applications accessing the VR HMD view and information simultaneously and so the post-processing component can be used to recreate the VR experience inside of Unity, shown in Image 9.

First a background Unity application is used to record all physiological data, and a generic screen recording software (such as OBS) can be used to record the VR view. Once a VR session is recorded, both the physiological data and OBS screen recording are used as input to the *PostProcessing* component. The OBS recording is rendered to a VR camera inside Unity, encompassing the entire field-of-view and physiological data is passed to the *DataProcessor* component as physiological service data objects. Here the data is processed in the same manner as real-time data including normalisation, filtering, and gaze pixel analysis.

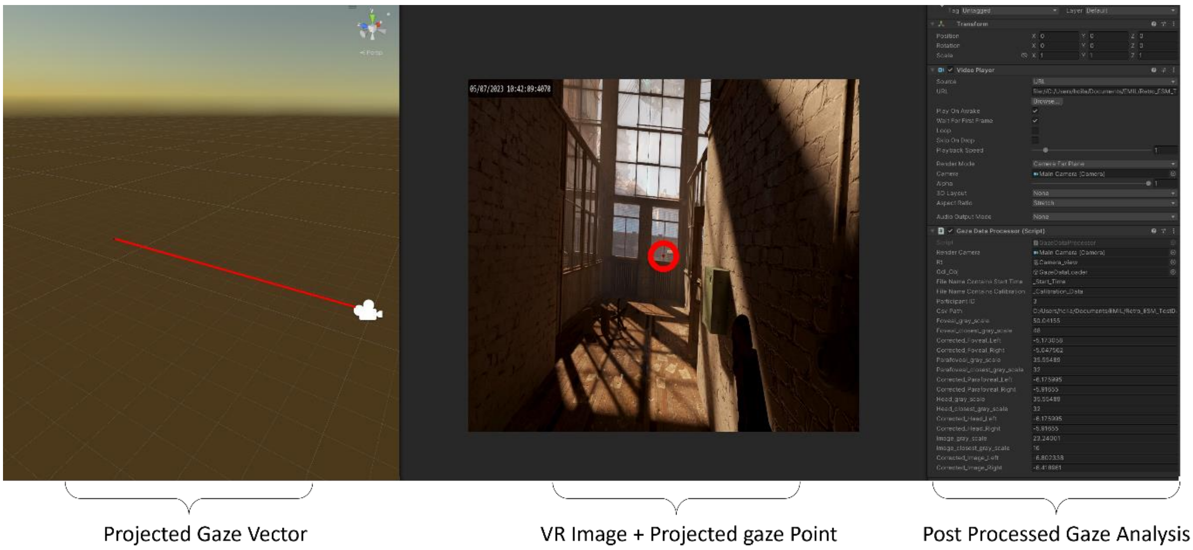

Image 9 An overview of post-processing procedure using an existing physiological file and OBS recording of a previous VR experience.

7 Conclusions

We have presented the *EmoSense* SDK for the integration of affect recognition in VR experiences. This SDK is currently maintained for the Unity game engine and is compatible with any VR hardware through the OpenVR and OpenXR standards that supports eye-tracking and pupillometry. The SDK allows developers to see predicted categorical and dimensional core affect and corresponding confidence intervals based on physiological measures both in real-time custom Unity applications, but also post hoc for conventional VR games and experiences. The SDK also provides two calibration programs for recording physiological and affect baseline measures and the document provides an overview of the calibration procedures, the recommended VR and sensor devices, and current external software dependencies. As a result, affect recognition using *EmoSense* can be incorporated into custom VR experiences developed in Unity enabling adaptive and emotion-aware content, and can also be used as a research tool for post hoc measures of User's evolving emotional states in existing VR games and experiences.