**Real-Time Arousal Detection Using   
Galvanic Skin Response Asset  
(Design Document)**

**Abstract**

The asset produces real-time features of Galvanic Skin Response (GSR) signal measured from particular player such as mean tonic and phasic activity level, rise time, recovery time, and slope of tonic activity. The asset will receive a filtered raw signal from a simple, low cost biofeedback device allowing sampling rate up to 1Khz. The results will be communicated from the server-side to the client component in order to be used for game adaptation. The level of arousal may be useful for emotion detection and for adaptation purpose, therefore the asset can be combined with the T2.3 Real-time Emotion Detection Asset and the T3.4 Player-centric rule- and pattern-based adaptation asset.

**Contact person**

Dessislava Vassileva, ddessy@gmail.com

Boyan Bontchev, bbontchev@fmi.uni-sofia.bg

**Table of content**

[- Description](https://rage.ou.nl/node/277#details_description)  
[- Asset architecture and input/output](https://rage.ou.nl/node/277#details_io)  
[- Examples of Use in Games](https://rage.ou.nl/node/277#details_useExample)  
[- Value](https://rage.ou.nl/node/277#details_value)  
[- Dependencies](https://rage.ou.nl/node/277#details_dependencies)  
[- Technical details](https://rage.ou.nl/node/277#details_techdetails)  
[- Licensing](https://rage.ou.nl/node/277#details_licensing)  
- References

**Description**

Galvanic Skin Response (GSR), also referred to Electro-Dermal Activity (EDA), Skin Conductance Response (SCR), Psycho-Galvanic Reflex (PGR), or Skin Conductance Level (SCL), is related to the activity of the sweat glands, which are regulated by the sympathetic nervous system [1]. When being open and functioning intensively, they emit water solution (sweat) which creates channels of higher conductivity toward the deeper skin layers [2]. EDA represents the electrical conductivity of the skin, which is directly dependent on the activity of the sweat glands, and is often used to index the autonomic arousal [3]. GSR offers a popular and affordable way for detection of player’s arousal in adaptive digital games and other affective computing applications.

The asset produces two main metrics featuring user arousal based on the GSR signal, namely:

* Current level of user arousal
* Current change in user arousal (increasing, decreasing or no changing) based on the measured GSR signal within the time window.

The level of arousal will be determined on a scale from zero up to maximum arousal level set by the asset user. This may be useful for emotion detection and for adaptation purposes, therefore the asset can be combined with the T2.3 Real-time Emotion Detection Asset and the T3.4 Player-centric rule- and pattern-based adaptation asset.

As well, the asset will be able to produce real-time features of GSR signal measured from particular player such as for example: mean tonic activity level, phasic activity represented by mean and maximum amplitude of skin conductance response (all in micro-siemens), rate of phasic activity (response peaks/sec), SCR rise time, SCR 1/2 recovery time, and slope of tonic activity (in micro-siemens/sec).

The asset will receive a filtered raw signal from a simple, low cost biofeedback device allowing sampling rate up to 1Khz. Measurements are carried out with two electrodes placed on two adjacent fingers. Recording, filtering and feature extraction might be executed on a computer (server) different than the game machine, in order to speed up all the required processing. The results will be communicated from the server-side to the client component in order to be used for game adaptation.

**Asset architecture and input/output**

Before starting with GSR measuring and arousal recognition, asset setup parameters should be initialised (by the client side). Such parameters include:

* required sample rate [Hz] – the frequency of measuring the GSR signal
* time window [s] – the length of time for calculating the moving average of the signal
* level of arousal [from 0 to N] – zero means no arousal at all and N means maximal level of arousal

The required sample rate [Hz] will be less than 1KHz. The calibration period [s], i.e. the time for baseline calibration, will be determined by experimental works and, next, set by the developer. Next, the client-side will expect receiving request for the indexed arousal level of the player and will reply to them in real time.

As well, he asset will be able to return values of several GSR features such as mean tonic activity level, phasic activity represented by mean and maximum amplitude of skin conductance response (all in micro-siemens), rate of phasic activity (response peaks/sec), SCR rise time, SCR 1/2 recovery time, and slope of tonic activity (in micro-siemens/sec).

Due to the intensive computing processing required for filtering and logging of the raw signal, as far as calculation of signal features, the asset is to be developed with a server and client side components communicating each other via sockets. Thus, the server component will be abre to reside at a computer different than the one running the game.

**Asset API**

* **int setSampleRate(int rate)** –
  + sets the frequency [in Hz] of measuring the GSR signal;
  + **rate** should be greater 1 and less than 100;
  + the default sampling is 10Hz; returns 0 in case of success
* **int setTimeWindow(int time)** –
  + the length of **time** [in seconds] for calculating the moving average, mean and standard deviation of the signal;
  + the default time window is 10s;
  + returns 0 in case of success
* **int setMaxArousalLevel(int level)** –
  + sets the max arousal level;
  + zero means no arousal at all and level means maximal level of arousal;
  + the value of **level** should be greater than 0 and less than 10;
  + the default value of the max arousal level is 2;
  + returns 0 in case of success
* **float getGSRFeature(string featureName)** –
  + returns the value (as float number) of a specific GSR feature;
  + **featureName** can be equal to **“average”**, **“movingAverage”**, **“standardDeviation”, “max”, “min”**;
  + returns -1 if the device is not started (active)
* **int getGSRLevel()** –
  + returns the value (as **int** number) of the current arousal level;
  + returns -1 if the device is not started (active)
* **int start()** – starts the device (after stop or pause); returns -1 if the device is not started (active)
* **int pause()** – pauses the device; returns -1 if the device is not paused
* **int stop()** – stops the device; all the buffers are flushed; returns -1 if the device is not stopped
* **int showGraphic()** – shows GSR signal graphic

**Design and Implementation Tasks**

1. Asset architecture and API design
2. GSR Device design and development
3. GSR Device functionality tests
4. GSR signal read and visualization
5. GSR features and arousal metrics calculation
6. Asset testing (coverage/unit/integration/performance tests)
7. Asset documentation (API doc, deployment instruction, tutorial, demo documentation)
8. Asset demo

**Examples of Use in Games**

Because of its simplicity and low cost techniques for measuring GSR (or EDA) continue to enjoy wide popularity - alone or combined with other techniques [4] to find correlations with the player experience and affective changes in player state. The most popular and simple stress detection algorithms based on EDA is proposed by Healey and Picard [5], without requiring a signal baseline value recorded in advance when the player is in maximal relaxation. The GSR baseline varies greatly in different people [6], which makes direct comparisons of GSR levels across individuals not possible. Because of this reason, Bersak et al [7] used the general trend of the GSR signal over a selected time period for calibration of relaxation in their “Relax-to-Win” two players racing game. Depending on the sign of the signal slope calculated by means of a least squares approximation, object’s speed was increased or decreased until reaching defined threshold values. EDA was successfully used together with other physiological measurements within an affective adaptation of the Pong game for implementing performance-based DDA [8]. GSR was used in [9] for stress detection by identifying rising and declining intervals of the EDA signal sampled every 0.5s. Car steering, speed, and road visibility were dynamically adapted in a car racing game [10] by analysing phasic skin conductance within a time window of 30 seconds.

**Value**

GSR or EDA provides rather easy and cheap measuring. Besides that, GSR is less ambiguous than facial muscles’ activity and heart rate [11]. As a disadvantage of GSR (i.e., EDA), it should be mentioned that possible external factors may change the humidity of the skin, and thus its electrical conductivity, which can lead to inconsistent results. EDA is influenced both by the values of temperature and humidity of the environment and the intake of food and refreshing drinks [12]. On the other hand, it must be taken into account that the reaction of EDA is not instantaneous and delays relative to the stimulus of 1 to 3 seconds.

Applying only EDA measurements for inferring emotional arousal and stress has a proven strong content validity as far as EDA is highly related to the arousal component of emotions [13]. However, using EDA in isolation for recognition of any emotion leads to a weak content validity. Nevertheless, player’s arousal serves as an useful basis for game adaptation purposes. Studies represented in [14] confirm both concentration and flow while playing video games increase EDA, which is an evidence for a combination of high arousal and pleasure.

**Dependencies**

There are no dependencies of this asset with the other assets. It can be used either in isolation or together the Real-time Emotion Detection Asset.

Naturally, asset use will be dependent on the GSR measuring device driver.

**Technical Details**

The asset will have only a client component. Its run-time requirements are as follows:

* Windows Vista or higher
* Microsoft .NET Framework

Source project requirements:

* Microsoft Visual Studio 2013 Professional to import the solution (project).
* Support for C#.

**Licensing**

No specific licensing is required.

**References**

1. Boucsein, W., 2006. Electrodermal Activity, 2nd ed. Springer, New York, NY, USA.
2. Andreassi, J.L., 2007. Psychophysiology, 5th ed., Lawrence Erlbaum Associates, Mahwah, NJ, USA.
3. Bach, D.R., Friston, K.J., Dolan, R.J. Analytic measures for quantification of arousal from spontaneous skin conductance fluctuations, Int. J. Psychophysiol. 2010 Apr; 76(1), pp.52–55. doi: 10.1016/j.ijpsycho.2010.01.011, http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2877802/
4. Broek, E. L. van der, Janssen, J. H., Westerink, J. H., Healey, J. A. Prerequisites for affective signal processing (ASP). Technical Report, 2009.
5. Healey, J.A., Picard, R.W., StartleCam: a cybernetic wearable camera, Proc. of the Second Int. Symp. Wearable Computers (ISWC 1998), IEEE Computer Society, Washington, DC, USA, 1998, pp.1–8.
6. Schwartz, M. S. Biofeedback: A Practioner’s Guide. New York: Guilford Press, 1995.
7. Bersak, D., McDarby, G., Augenblick, N., McDarby, P., McDonnell, D., McDonald, B., Karkun, R. Intelligent biofeedback using an immersive competitive environment. Designing Ubiquitous Computing Games Workshop, UbiComp, Atlanta, USA, 2001, September.
8. Liu C., Agrawal P., Sarkar N., Chen S. Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback. Int. J. Hum. Comput. Interact., 2009, 25, pp.506–529.
9. Chittaro, L., Sioni, R. Affective computing vs. affective placebo: Study of a biofeedback-controlled game for relaxation training, Int. J. of Human-Computer Studies, Vol. 72, Iss. 8–9, August–September 2014, pp.663–673, doi:10.1016/j.ijhcs.2014.01.007
10. Parnandi, A., Gutierrez-Osuna, R. A comparative study of game mechanics and control laws for an adaptive physiological game. J. on Multimodal User Interfaces, 2014, pp.1-12.
11. Kivikangas, J. M., Chanel, G., Cowley, B., Ekman, I., Salminen, M., Järvelä, S., Ravaja, N. A review of the use of psychophysiological methods in game research. Journal of Gaming & Virtual Worlds, 2011, 3(3), pp.181-199.
12. Pflanzer, R., McMullen, W. Galvanic Skin Response and the Polygraph. BIOPAC Systems, Inc., http://www.vhlab.umn.edu/~bmen3701/Documents/Biopac9.pdf
13. Fairclough, S. H. Fundamentals of physiological computing. Interacting with computers, 2009, 21(1), pp.133-145.
14. Giakoumis, D., Vogiannou, A., Moustakas, K., Kosunen, I., Devlaminck, D., Ahn, M., Burns, A.M., Khademi, F., Tzovaras, D. Multimodal monitoring of the behavioral and physiological state of the user in interactive VR games, eNterface’09 Workshop, July 13th – August 7th, Genoa, Italy, 2009, http://www.iti.gr/~moustak/eNTERFACE09\_Bio.pdf