

A Concept for Visualizing Psychophysiological Data in Human Computer Interaction: The FeaturePlotter

Falko Pross^(✉), Dilana Hazer, Harald C. Traue, and Holger Hoffmann

Medical Psychology, University of Ulm, Ulm, Germany
falkopross@googlemail.com, {dilana.hazer,
harald.traue,holger.hoffmann}@uni-ulm.de

Abstract. This paper introduces a graphical concept and an implementation for visualizing psychophysiological data in human computer interaction. Psychobiological measurements result in huge datasets, which are mandatory for the development of semi-automatic or automated emotion classification and hence a reliable planning and decision-making system called companion system. The mentioned amount of data calls for the need of making dependencies and coherences in those datasets visible for the human eye in addition to algorithmic pattern recognition and feature selection. Seeing through the data by exploring it playfully helps experts understanding the data structure and provokes non-specialists' curiosity.

Keywords: Data visualization · Psychophysiology · Companion systems · Emotion recognition · Human computer interaction

1 Introduction

Technical devices are getting faster and smaller by the week and computers affect the way we live more than ever. Invented as highly specialized machines and suitable for expertized operators only they migrated into our bags and pockets. They assist us in our daily routine and therefore must be able to perform a multitude of completely different functions. Hence the interaction between humans and machines becomes more complex and versatile, despite the steady enhancements by the efforts of user interface design and the technical development of more and more intuitive interaction possibilities like for example multitouch input.

This makes new steps in the evolution of interactions between humans and computing machinery possible – an evolution towards companion systems. Knowledge about the users' properties, current physical and emotional conditions and the environment, allows companion systems to anticipate the users' needs and wishes and to simplify the interaction by reducing its complexity [1]. Companion systems are reliable planning and decision-making systems depending on a large set of data. Especially the process of the analysis of naturalistic user behavior - which is often based on a multimodal approach combining the analysis of prosody, mimics, gestures and psychobiological measurements - has to deal with great quantities of data [2].

2 Materials and Methods

2.1 The Emergence and Structure of Feature Data

Psychobiological measurements depend on an elaborate setup and are therefore realized in controlled experimental settings or measured in quasi-experimental observational settings, during which a number of independent variables, like for example a variety of emotions or different pain levels are induced [3]. Specialized hardware is measuring body functions and storing the values permanently over the course of time. Measured body functions are mainly skin conductance, blood volume pulse, electrocardiography, electroencephalography, electromyography or respiration.

During multiple processing steps a subset of the collected data is created: First the data is cut, so only the important sections shortly before and after the induction of an independent variable remain. Next the data is filtered to remove or reduce noise and undesirable frequencies [3]. Finally features are extracted from the preprocessed data, by performing a variety of mathematical functions on the dataset (e.g. amplitude, frequency) [4, 5]. The results are tables containing the feature values for each subject and measured point of time categorized by the inducted independent variable of that point of time.

In the further process of semi-automated emotion classification a subset of the feature data is used in classification algorithms like neural networks, k-nearest neighbor or support vector machines [6]. These algorithms train with part of the data and should afterwards be able to classify given datasets into independent variables. The success of correct classification highly depends on the applied subset of features.

The selected feature dataset is often more confusing than the initial amount of data, so it is still way too complex to be interpreted by humans and hence not possible for humans to choose the perfect set of features. For example there is one table for each subject – which means it is necessary to merge and compare a multitude of tables, which might be mentally stressing lots of cognitive capacity. In fact, not all the extracted features are relevant for the analysis. There are guidelines describing which features are suitable for which purposes and there are methods of algorithmic feature selection optimization [7]. Both aids are useful and necessary; nevertheless they still do not offer a possibility to understand the data directly.

2.2 Data Visualization

“Often the most effective way to describe, explore and summarize a set of numbers – even a very large set – is to look at pictures of those numbers” [8]. The previous sentence outlines the need for a visualization of the described feature data. Data visualization can be considered as external cognitive resources extending the human brain.

Following the state of the art and the proceedings in computer graphics as well as the rapid growing amounts of data, new approaches in data visualization have evolved over the last decades: Instead of specifying the entire representations of the data and its shapes, positions and colors, only a set of rules is determined [9]. These rules allow the

user exploring the data actively and thus getting more involved. Important aspects for data visualization, meant to arouse interest and curiosity are: aesthetics, efficiency, information and novelty [10].

2.3 Graphical Concept

For visualizing multiple connected datasets in an easy and well structured way, a graphical concept based on the simplicity of a bar diagram has been developed. It starts with reducing the bars to simple points and then displaying them in one common scale (see Fig. 1). This way it is possible to consolidate all results of one subject, one feature and one independent variable in one figure.

Of course there exists way more than one feature. Also, it is important to compare the values of features with each other, so for the sake of intuitive visual comparison they have to be placed as close together as possible. To minimize the distance between an arbitrary amount of scales, they are arranged circular. And to keep track of a lot of small values, the null points of the scales are not unified in one point, but stretched apart in a circle (see Fig. 2).

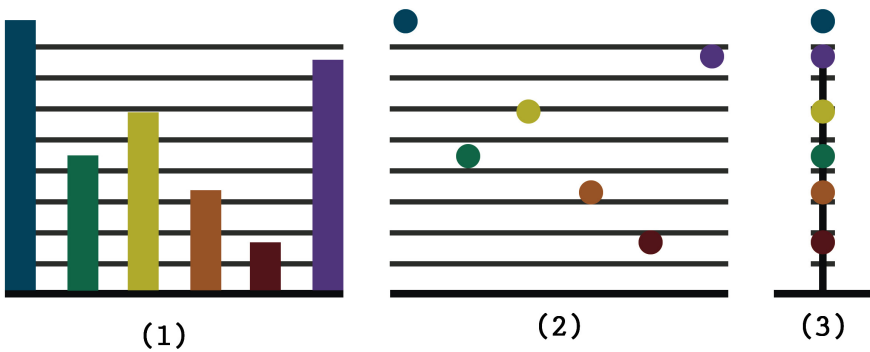


Fig. 1. Progression from bar diagram (1) to a simplified bar diagram (2) to a scale (3)

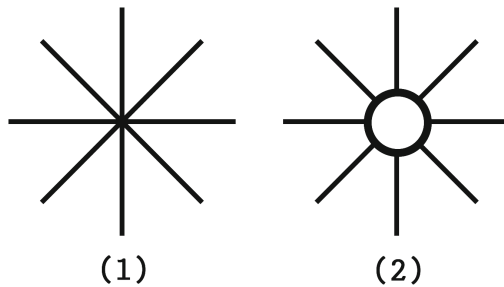


Fig. 2. Circular arranged scales (1) and scale circle with inner null circle (2) for better overview

Now each of those scale circles visualizes the values of all features of one subject and one independent variable, which provides a good overview over the feature values and makes it possible to answer the first question concerning the selection of features and the further processing towards classification and semi-automatic emotion recognition are:

1. Are the values of one feature, one subject and one independent variable all within a certain range or are they distributed randomly along the whole scale?

Randomly distributed values along a scale signify a feature which is not showing any methodic tendencies, like for example low values for an independent variable A and high values for an independent variable B. If the values of a feature do not show methodic tendencies, they can not be compared to other values. So the questions above answers if a feature is reliable and principally suitable for classification.

Further, there are two more important issues:

2. Do feature values of a subject vary among different independent variables? At which features do they vary? How do they vary?
3. Do feature values of an independent variable vary among different subjects? At which features do they vary? How do they vary?

It is not possible to answer these questions by use of the current form of representation, so the visualization concept has to be extended by another dimension: By arranging multiple scale circles on top of each other, feature values of either different independent variables (question 2) or different subjects (question 3) can be compared directly (see Fig. 3).

By adding a third dimension to the visualization, it is possible to change from the front view to the side view. This makes it possible to compare specific features in detail (see Fig. 4).

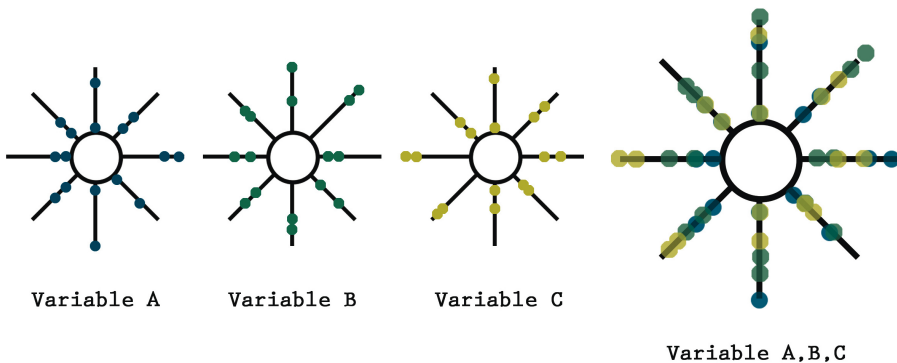


Fig. 3. Multiple scale circles on top of each other can be compared directly

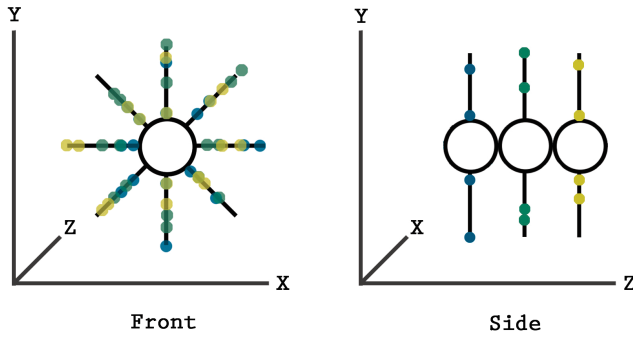


Fig. 4. Multiple scale circles on top of each other from front view and from side view

2.4 Fields of Application and Related Conditions

The FeaturePlotter application is not meant to be a static tool exporting static graphics. It is designed to create an interactive environment where it is possible to explore scientific data playfully. Therefore the data is visualized in a 3D-model, which can be manipulated in real-time by the user. Besides providing different perspectives and hence more possibilities of understanding the data altogether, the interaction involves the user and encourages to go beyond known boundaries.

Considering the qualities mentioned above, there are multiple fields of applications for the FeaturePlotter: Of course first of all it is a scientific tool for experts, helping them to understand data features. But it is also a tool for presenting, explaining and illustrating new or old feature data in front of an audience, for example in lectures or conventions. Furthermore, prearranged views of the data can be exported into static images and used in reports, documentations and journals. And on a final note it is a possible door opener in conversations with non-experts - for example at exhibitions or open house days: The application can arouse curiosity through its graphical design and its interactive capabilities.

2.5 Conditions

To meet all the expectations resulting from the fields of application, several conditions had to be considered during the implementation: Platform independence and an easy installation and handling are important features for providing a ready to use solution for displaying data without the need of having expert computer skills. This way it is possible for almost everyone to use the application on multiple different devices to show visualized content and share knowledge about it.

An innovative and attractive design arouses the curiosity of non-specialists and invites them to explore the graphical structures, shapes and colors. This is where an easy handling and detailed instructions and explanations guide a non-expert user through the application and introduce him to the topic and the context of the visualization.

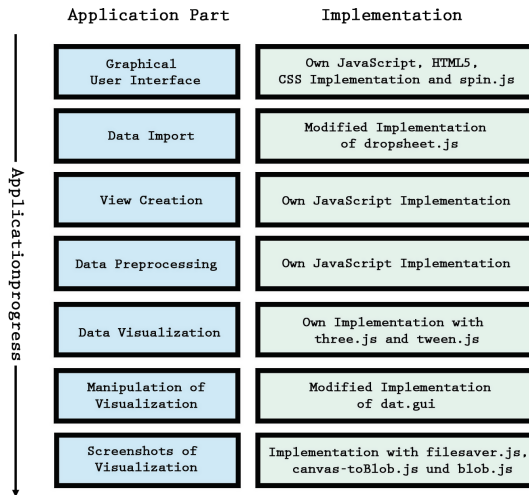


Fig. 5. Application structure and overview

2.6 Implementation

Using web technologies like HTML5, CSS, JavaScript and additional JavaScript libraries, all of the described conditions can be implemented smoothly. The application is developed as a web app, which is similar to a standard website but with an additional functionality. It is possible to run an instance of the application in every common browser without an installation or setup – just by visiting the corresponding domain online.

The application implementation is based on single page architecture: There exists only one HTML-File, which is dynamically filled with content via JavaScript. This avoids loading content from the server subsequently after each interaction and therefore increases the performance.

Additional to the browser version, there are desktop versions of the application for offline use on Mac OS and Windows. A similar version of the web app has been packed into a node-webkit container to provide server functionality for the web technologies without having to run a server locally. In Fig. 5 an overview over the structure and different parts of the application and used libraries is illustrated.

3 Features

3.1 Graphical User Interface

Because the application is not meant to be a scientific tool only, but a simple and inviting application for non-experts as well, the graphical user interface is structured in plain lucid sections and furnished with big self-explaining buttons and simple supervising texts (Fig. 6). In addition, there is a tooltip function explaining every

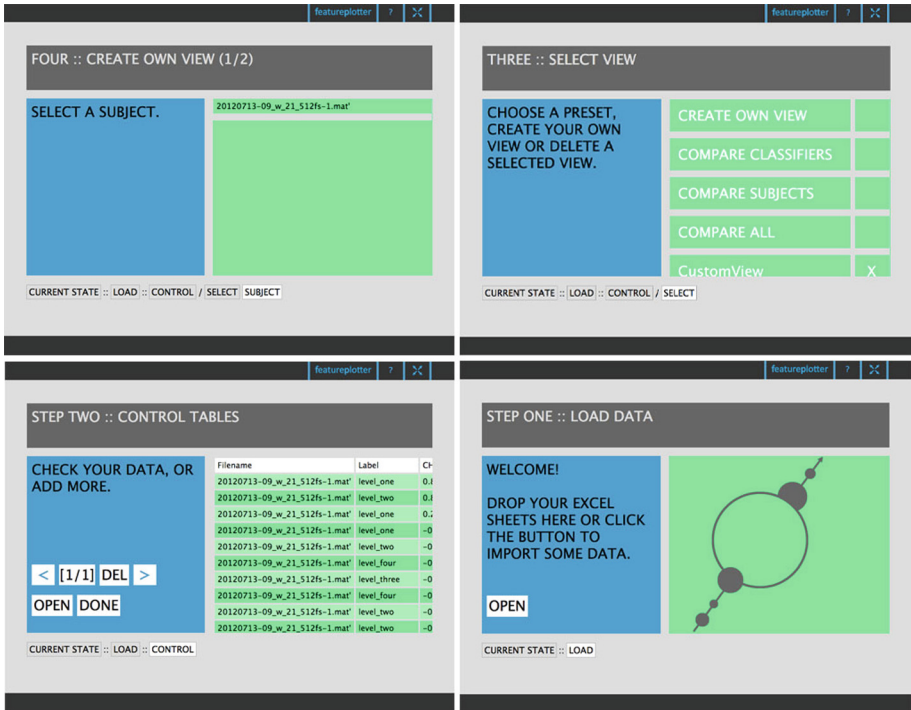


Fig. 6. Examples for the graphical user interface

button in the application. At the bottom of the application, there is an easy to use breadcrumb navigation showing the current state of the application and allowing to redo previous steps.

3.2 Data Import

The first step in the application is to import the feature data, which usually is available as CSV files or excel-sheets. The application allows to simply drag and drop one or more excel-files onto a denoted area, or to open the operating system’s file explorer by a button. Afterwards the tables are parsed and converted into JavaScript-Arrays. The imported data is displayed in tables to enable the user to check if everything has been imported correctly.

3.3 Views

As mentioned before, a scale circle is created from a combination of a subject and a variable. In the second step of the application it is possible to choose from different predefined views, where combinations of subjects and variables are already picked,

to create an own view by selecting combinations of subjects and variables or to pick a previously created view.

There are three predefined views: “Compare all subjects”, “Compare all classifiers” (variables) and “Compare all”. For creating an own view the user can select a subject and a variable for each requested scale circle.

After selecting pairs of subjects and classifiers, a corresponding subset of the imported data is created. Then the peaks and lows, the average and the standard deviation of each feature are calculated to normalize the values and wipe out discordant values.

Of course custom views can be stored permanently. This is implemented by using the localStorage-API of HTML5. Before a custom view is stored, it is converted into a single string object. When a view is loaded, the corresponding string is parsed and converted back into a view object.

3.4 Visualization

To explore the data from different points of view the application allows to move freely in 3D space by rotating, zooming and panning the camera around the visualization elements. The 3D scene is implemented with three.js, a graphics library for JavaScript based on WebGL. Multiple selectable mouse-over effects and text labels help to identify and examine individual scale circles, features and data points. In the visualization, data points which are close to each other grow bigger to improve the visual illustration and emphasize similar measurements. The scaling factor can be changed by the user as well as the minimum distance between two points specifying if they are being scaled or not (see Fig. 7).

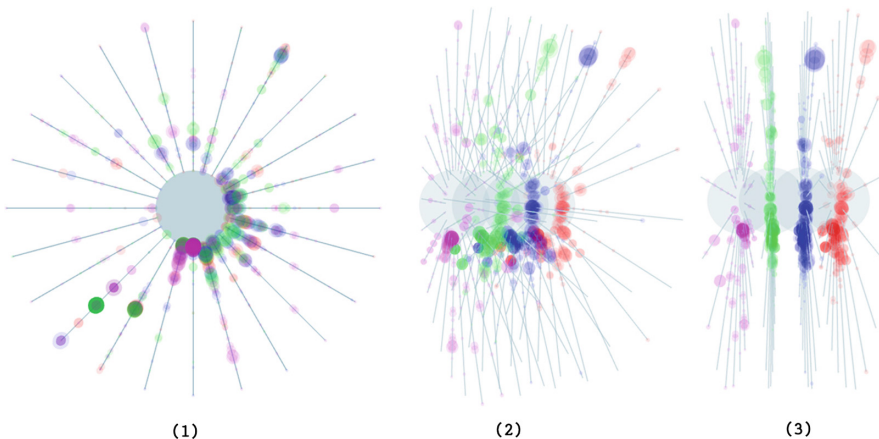


Fig. 7. Screenshots from the application showing a camera rotation from front view (1) to side view (3).

4 Restrictions and Suggested Improvements

The application performance is dependent on the number of elements to be visualized in the three-dimensional scene. Therefore it is not possible to visualize an arbitrary number of scale circles at once. The graphical concept is not meant for the visualization of many elements as well. It is designed to compare a preselected set of subject-variable-combinations. Of course improvements in the development of WebGL will improve the performance of the FeaturePlotter-Application too.

Especially on mobile devices the performance is not ideal, due to additional multiple browser-dependent problems: Some mobile browsers have problems displaying the application correctly when rotating the device or zooming in and out of different sections of the screen. Optimizing the application in terms of responsiveness on different devices, mobile platforms, screen sizes and computing power is necessary to enable a fully functional and stable mobile version of the application.

Data cannot be stored inside the application, so it has to be imported every time the application is started. In combination with customized views this can cause inconsistency when varying datasets have been imported. Storing the data inside the application is not possible regarding the underlying web technologies and their security restrictions. A simple way to simplify the data import and reuse of already imported data would be to store the path of already used data files locally and ask the user if certain datasets – which are required for the available or selected views – should be imported.

The formatting of the data tables is predefined and has to be strictly maintained to ensure correct import and further processing.

5 Conclusion

The FeaturePlotter-Application is the first iteration of a multifunctional tool helping people to overview, compare and understand big data sets with an intuitive visual data screening. It allows interacting with present data, exploring it playfully and discerning coherences and differences. Its simple, respectively unnecessary installation and its simple structure and design provide everyone with the possibility to use the application straightforward without the need of having experience in difficult scientific tools, complex programs or even the subject matter itself.

Of course the application has to be tested in the daily routine of experts and non-experts and there the actual benefit will become apparent as well as the need for further development, additional functionality or the redesign of specific parts of the application. There are already plans to add other modules and functions like for example animating the positions of the data points to illustrate the progression of the values over time during an experiment. So the FeaturePlotter-Application can be considered as a starting point for different researches and studies in the field of psychophysiological data visualization.

Acknowledgements. This research was supported by grants from the Transregional Collaborative Research Center SFB/TRR 62 Companion Technology for Cognitive Technical Systems funded by the German Research Foundation (DFG).

References

1. Wendemuth, A., Biundo, S.: A companion technology for cognitive technical systems. In: Esposito, A., Esposito, A.M., Vinciarelli, A., Hoffmann, R., Müller, V.C. (eds.) COST 2102. LNCS, vol. 7403, pp. 89–103. Springer, Heidelberg (2012)
2. Walter, S., Scherer, S., Schels, M., Glodek, M., Hrabal, D., Schmidt, M., Böck, R., Limbrecht, K., Traue, H.C., Schwenker, F.: Multimodal emotion classification in naturalistic user behavior. In: Jacko, J.A. (ed.) Human-Computer Interaction, Part III, HCII 2011. LNCS, vol. 6763, pp. 603–611. Springer, Heidelberg (2011)
3. Walter, S., Gruss, S., Limbrecht-Ecklundt, K., Traue, H.C., Werner, P., Al-Hamadi, A., Diniz, N., da Silva, G.M., Andrade, A.O.: Automatic pain quantification using autonomic parameters. *Psychol. Neurosci.* **7**, 363–380 (2014)
4. Cao, C., Slobounov, S.: Application of a novel measure of EEG non-stationarity as “Shannon- entropy of the peak frequency shifting” for detecting residual abnormalities in concussed individuals. *Clin. Neurophysiol.* **122**, 1314–1321 (2011)
5. Chen, W., Zhuang, J., Yu, W., Wang, Z.: Measuring complexity using FuzzyEn, ApEn, and SampEn. *Med. Eng. Phys.* **31**, 61–68 (2009)
6. Hsu, C.-W., Chang, C.-C., Lin, C.-J.: A practical guide to support vector classification. *BJU Int.* **101**, 1396–1400 (2008)
7. Kolodyazhnyi, V., Kreibig, S.D., Gross, J.J., Roth, W.T., Wilhelm, F.H.: An affective computing approach to physiological emotion specificity: toward subject-independent and stimulus-independent classification of film-induced emotions. *Psychophysiology* **48**, 908–922 (2011)
8. Tufte, E.R.: *The Visual Display of Quantitative Information*, 2nd edn. Graphics Press, Cheshire (2001)
9. Hidalgo, C.A., Almassawi, A.: *The Visualization Revolution* (2014). <http://www.scientificamerican.com/article/the-data-visualization-revolution>. Accessed 02 February 2015
10. Fry, B.: *Visualizing Data*. O’Reilly Media, Sebastopol (2008)