УДК 378:004
DOI: 10.25686/978-5-8158-2474-4-2025-498-504

A unified approach for EEG visualization on a mobile device

Ivan A. Labutin
Perm State University, Perm, Russia

Abstract. In this paper we propose a concept of applying unified methods for EEG visualization on a mobile devices based on the ontologically driven software generation pipeline. The importance of this work comes from the rapid development of wearable hobbyist-class neurocomputer interfaces and the request for the unified approach for bringing such devices and mobile gadgets together for the purpose of EEG visualization and analysis. This paper reviews existing solutions for visualizing EEG on mobile devices, touches the properties of Aurora OS and its graphics stack in application to visualization. The software solution based on an ontology-driven architecture presented in our previous work is employed to solve a problem of automatic generation of application code. This application is deployed and tested on a real hardware device running Aurora OS 4.0, further validating our approach.
***Keywords*:** neurointerface, BCI, ontology engineering, IoT

Introduction

There is a growing tendency of moving EEG research and experiments from the walls of medical institutions into more personalized and lax environment. Modern mobile devices are more than capable of collecting and processing EEG data. [1] notes that EEG pattern discovery can be beneficial in a wide spectrum of possible applications in natural conditions, such as sports, education, entertainment, rehabilitation, enriching traditional approaches. In last decade a few modern mobile EEG systems have emerged to fight the limitations imposed by a traditional clinical equipment, but their widespread employment still remains restricted due to interoperability issues. [2] identified more than 900 studies that employed numerous hobbyist-grade EEG devices (Emotiv Epoc, NeuroSky, Muse etc.) that shows a growing trend for interest in mobile EEG research. Due to this fact, it may be concluded that the task of developing unified methods for visualizing EEG signals on smartphones that became our everyday companion devices is of utmost importance.

In our previous work [3] we proposed a unified ontology-driven approach to describe all aspects of an interaction with BCI, generate a code of a smart mediator that can be used to query data from an EEG headset and preprocess it. Such a smart mediator may manifest itself as a mobile application, allowing for quick integration of the neurocomputer interface with different experimental settings and devices. This paper examines the relevance of mobile EEG visualization, analyzes existing developments on this topic and features the implementation of a mobile EEG real-time visualization application employing described approach.

Relevant Studies

In recent years, mobile devices with powerful built-in sensors have opened up new possibilities for neuroscientific experiments outside of rigid medical environment. Some researchers note [4] that modern smartphones put computing power and data collection tools in your pocket, which is why there is a demand for portable EEG recording systems that can function outside of a laboratory. Indeed, commercial mobile and hobbyist-grade EEG devices have appeared (for example, Emotiv EPOC, Neurosky MindWave, Muse and others), combining compact headcaps with wireless data transmission to a smartphone or tablet [5]. For example, [6] describes a development of an 24channel mobile EEG system DreamMachine with a companion Android application EEGDroid,

which provides visualization of the signal in real time. The application offers scaling, enabling or disabling individual channels, choosing the gain and filters, and provides a live monitoring of the recording quality.

A number of studies have been focusing on describing the transfer of classical EEG analysis methods to mobile platforms. For example, in the work [4], the Smartphone Brain Scanner project was introduced. It is a system that combines a wireless 14-channel EEG headcap and a smartphone for the purpose of a 3D visualization of brain activity in real time. The authors used Qt framework, suitable for cross-platform development. Despite the lower signal quality compared to stationary high-density systems, the authors point out that mobile approach opens up tremendous new possibilities for neuroscientific research in non-stationary environment. However, the provided implementation had some disadvantages: the system required a rooted smartphone and a specially prepared kernel, and the issues of delays and synchronization of stimuli required additional investigation.

In the work [7], a modular approach to creating a BCI applications on Android was proposed. They implemented the SCALA (Signal ProCessing and CLassification on Android) application within the framework of a “multi-app” architecture, where separate modules for stimuli presentation, EEG capture, online processing and classification are different apps that simultaneously launched on the smartphone, and the data is transmitted between them via TCP/UDP network connections. The system turned out to be quite flexible: according to the authors, it can be extended with different EEG devices. Their experiments confirmed that timing is not a problem - a system can achieve near-laboratory accuracy, but there are some issues related to OS task scheduling and device performance that can affect it.

Another approach, training neural network models to control a smartphone interface, is demonstrated in [8]. It presents a prototypical iPhone application that recognizes P300 potentials and blinks from EEG signals, allowing a person to control a phone dial interface.

There are also specialized mobile applications for educational or auxiliary purposes. For example, [9] created iBraiN2 and iBraiNEEG2 applications for Android and iOS, which can visualize brains’ functional connectivity networks and the arrangement of electrodes according to the 10–20 system on a 3D transparent model of the cortex. These applications were created in C# using the Unity game engine. The authors emphasize that their solutions are the first embedded mobile applications for interactively displaying functional networks of the brain. The main goal of iBraiN2/ iBraiNEEG2 is educational: they allow doctors and students to study the anatomy and connections in the brain in an interactive mode. However, these programs work with pre-calculated atlas and network data and do not connect to real EEG devices; that is, they do not perform signal analysis by themselves, but only demonstrate prerecorded information.

An important aspect in developing mobile EEG applications is the employed graphical user interface framework. For example, as was already mentioned, Smartphone Brain Scanner used Qt for cross-platform implementation and was able to provide 3D model display at 30˜ frames per second and smooth touch control [4]. The subsequent version of this system (SBS2) is also implemented with Qt/C++ and runs on Windows, Linux, Android and other systems [5]. At the same time, some other applications are built using relatively high-level technologies – for example, the aforementioned iBraiN/iBraiNEEG is based on the Unity engine and C# [9]. The choice of lightweight GUI components in regards to a mobile device seems logical due to the limited resources of the device: interfaces should be as simple as possible and not take away computing power. For example, the EEGDroid application provides only a few basic controls and a graph view to ensure a lightweight and comprehensible visualization of EEG without unnecessary bloat [6].

There’s a relatively distinct area of research that attempts to standardize the description of neuroscientific experiments, their methods, tools and data properties. This embodies projects with the aim to create ontologies for EEG, for example NEMO (Neural ElectroMagnetic Ontologies) [10] or BCI-O (Brain-Computer Interface Ontology) [11]. Both of them have the goal of formalizing the concepts of signals, episodes, stimuli, etc. Ontology-driven software generation could facilitate the creation of specialized brain-computer interfaces (e.g., by automatically linking UI elements to signal semantics), and this was demonstrated in our previous work [12].

To summarize, the field of EEG visualization on mobile devices demonstrates the possibility of transferring many functions of neurocomputer interfaces to portable applications [13, 9]. At the same time, current mobile solutions are still limited in signal quality and versatility: low-density recording and consumer sensors produce more noise [13], and smartphone operating systems have delays that affect the accuracy of timestamps [7]. Most existing applications perform auxiliary, demonstration or educational functions and do not claim to be competitive with the classic laboratory-grade systems [9]. In particular, there are still no signs of widespread use of unified, adaptive systems capable of automatically generating an interface for various types of experiments. The proposed ontology-driven approach is intended to close this gap by providing a single conveyor from the description of the experiment to the final mobile application.

Suggested Solution

The proposed solution is based on the ideas from our previous works [14, 15, 12, 16], which formulated an approach of ontology-driven integration of neural interfaces into an IoT infrastructure. The main essence of this approach is the introduction of a smart mediator between the neurocomputer interface and other IoT components, with its firmware being automatically generated based on a control ontology compiled from a set of ontological descriptions of the environment, data transformation modules, and the neural interface itself [14, 15]. This ontology describes the characteristics of all data transfer and transformation parameters, which allows for automatic construction of interaction pipeline. We also proposed [12] a mechanism for constructing a pipeline that combines ontological descriptions of smart mediator components through the interface of the SciVi scientific visualization and visual analytics platform [17, 16, 18, 19], where the researcher describes the desired components and their interconnection, and the ontologically controlled generator automatically creates the mediators' firmware [15].

In this work, we expanded the available toolkit of our system with the „EEG Visualize" operator, which can be included in the smart mediator to perform visualization of raw EEG data on the screen of a mobile device in real time. This operator is a separate part of the data processing pipeline, similar to the „Sliding Window" or „Montage Provider" we previously introduced [16]. At the stage of creating the firmware of the smart mediator, this visualization module is included in the code generation process in order to embed into the mediator with the purpose of receiving and displaying „raw" EEG signals.

The key advantages of the proposed concept are that it is based on proven principles of ontology-driven integration [14, 15], thus automating the inclusion of the visualization module into the system without a need for manual code creation, providing unification (the EEG device driver and other modules can be easily swapped without manual modification of the mediators' code [16]) and allowing to employ a mobile device as a platform for EEG data visualization.

As a result of the code generating, the final firmware of the smart mediator contains a built-in EegVisualize module, which draws an EEG graph on the display in real time. This approach is consistent with the ideas of earlier publications, where it was noted that the approach for BCI-IoT interoperability should support the integration of neurocomputer interfaces without deep knowledge of neuroscience or coding skills [14, 15]. With our method, the researcher employs high-level tools to solve his specific research problem („visualize EEG signals"), and the system itself turns the pipeline he built into a firmware code.

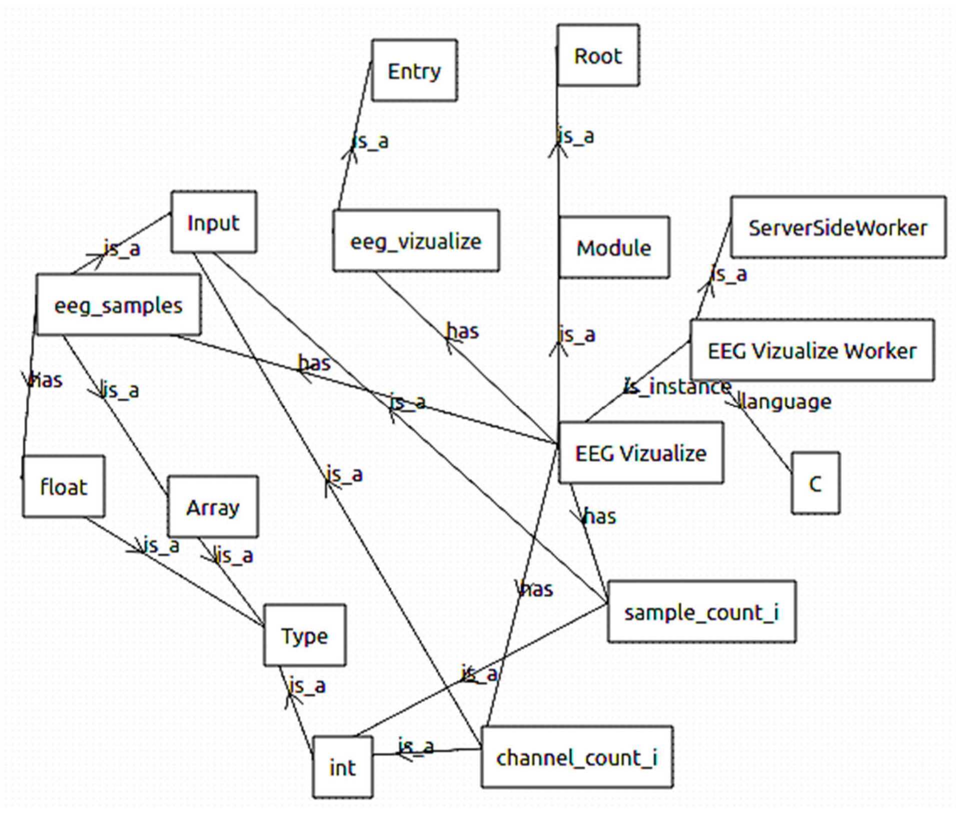

Figure 1. „EEG Visualize" operator ontology

Testing and Validation

Д To verify the viability of the concept, the new „EEG Visualize" (Fig. 1) operator was incorporated into a data processing pipeline (Fig. 2) employed to run raw EEG visualization on the Aquarius NS220 tablet device running Aurora OS. This device is equipped with a high-quality 10.1-inch IPS screen with a resolution of 1920×1200, which ensures a clear view of fine details of EEG graphs. In the experimental setup, an external neurocomputer interface EBNeuro BE Plus LTM was used, connected to the tablet via Wi-Fi network. Signal acquisition rate was set to 512 Hz, and we used 19-channel configuration (21-electrode headcap). The assembled mediators' code including the implementation of visualization operator is loaded onto the tablet in form of a mobile application, and during its operation, raw EEG data are received by it in real time via the constructed pipeline.

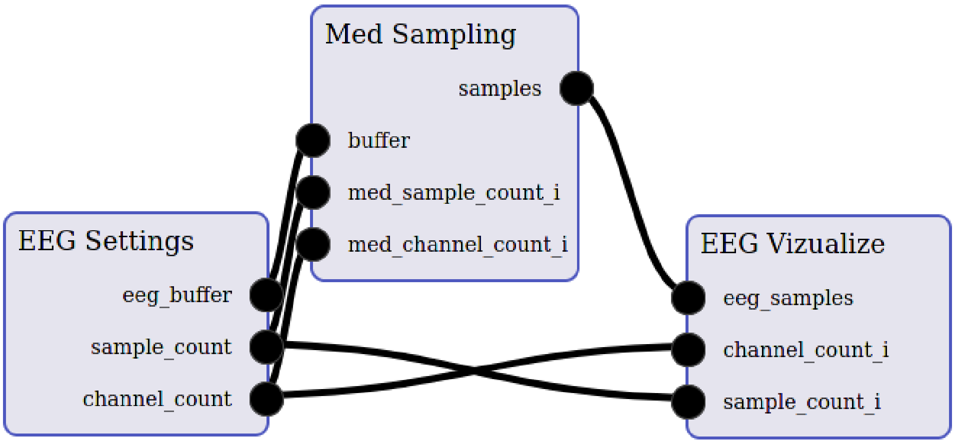

Figure 2. An overview of a pipeline

Aurora OS is a Russian mobile OS based on the Linux kernel and the libhybris compatibility layer, which allows using Android drivers in GNU/Linux environment. It employs the Wayland/Qt stack for graphics and Silica's own QML components for the user interface. Applications for Aurora OS are developed using Qt Quick (QML/JavaScript) and, if necessary, C++. The user interface is created declaratively in QML, and the application logic can be implemented in C++ or JavaScript. This architecture allows for a nice balance between a hardware-accelerated rendering and smooth operation of the interface on mobile devices and an ease of programming.

The visualization operator introduced in this paper consists of both C++ and QML/JavaScript code. It uses the basic Canvas class for displaying EEG signals as polylines without any smoothing. Screen update is performed at 30 frames per second regardless of the signal acquisition rate. Each channel is drawn using a distinct color from a predefined palette. The testing process included the following steps:

1. Creation of firmware for the smart mediator. The SciVi editor was used to describe the sequence of operators (Fig. 2). The „EEG Visualize" operator is added to the diagram at the end of the flow. After that, the glue source code combining the modules of the smart mediator is generated based on the ontological description of modules and the created pipeline (Fig. 3).
2. Deployment to the device. The compiled mediator application was installed on the Aquarius NS220 tablet. When launched, the application establishes a connection with the neurocomputer interface and begin signal acquisition.
3. Visualization examination. The tablet screen displays the raw EEG signal graph (Fig. 4) in real time.
4. Performance evaluation and estimation. During operation, graph update delays were visually estimated. No noticeable delays were spotted, which confirms the suitability of the solution for real-time applications.

During testing, the application was launched on the real device, receiving an EEG stream from EBNeuro BE Plus LTM EEG amplifier. After the launch, the application initializes the EEG amplifier driver, starts to receive the data and displays signal curves as stripes on the screen, updating them as the new data arrive. The interface is really simplistic and offers no governance over the process, but it may be extended to incorporate control widgets: channel selection, scaling, stopping the stream, etc. In practice, it was observed that the QML/Qt graphics engine is capable of drawing several dozen EEG channels without delay, providing smooth animation of graphs. It may be beneficial to add a way for the user to interact with widgets via the touch screen: for example, provide the ability to scale the time window. The application correctly responds to background events (minimizes, continuing recording) thanks to the built-in Aurora OS life cycle management mechanisms.

```
float* eeg_buffer;
int channel_count;
int sample_count;
float* samples;

void eeg_settings(float** eeg_buffer, int* channel_count, int* sample_count);
void med_sampling(float* buffer, int med_sample_count_i, int med_channel_count_i,
float** samples);
void eeg_vizualize(float* eeg_samples, int channel_count_i, int sample_count_i);

eeg_settings(&eeg_buffer, &channel_count, &sample_count);
med_sampling(eeg_buffer, sample_count, channel_count, &samples);
eeg_vizualize(samples, channel_count, sample_count);
```

Figure 3. Listing of generated glue code

During the tests, the Aquarius NS220 tablet demonstrated the convenience of the platform: a stable, uninterrupted connection to the neurocomputer interface via Wi-Fi, robust operation of the OS, and excellent high-quality graphical output to the screen. Thus, it was confirmed that the „EEG Visualize" operator functions correctly as part of a smart intermediary, and the selected equipment is a suitable platform for such experiments. Most importantly, our solution [12, 16] proved to be applicable for the task of real-time visualization of EEG data collected in the IoT environment on the client device.

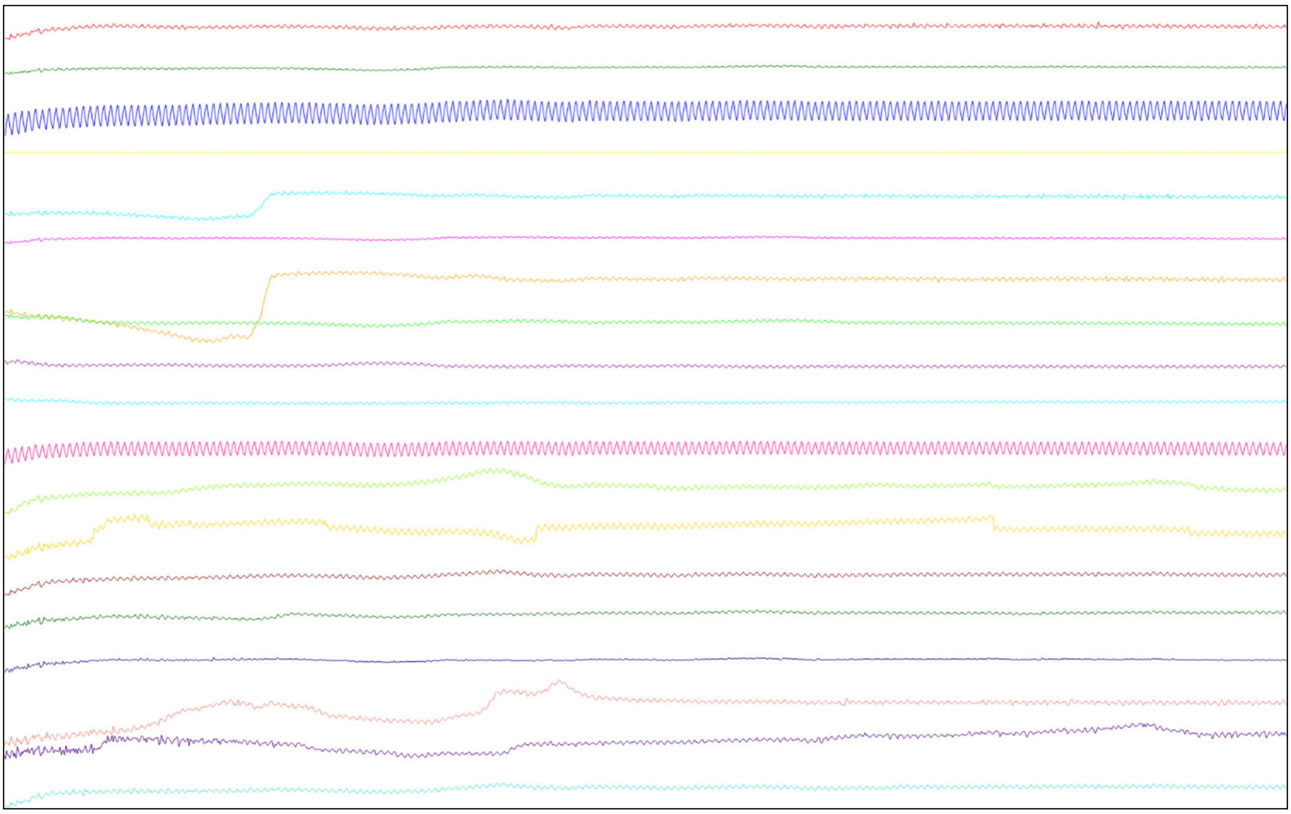

Figure 4. Application screenshot displaying visualization of an EEG signal.
Different EEG channels are represented with different colors

Conclusion

The paper reviews modern approaches to visualization of EEG data on mobile devices. A review of existing mobile applications and tools for working with EEG data is conducted. An ontology-driven solution is employed that simplifies the creation of an EEG data visualization application. Aurora OS-powered Aquarius NS220 tablet is used as a platform. Testing showed the applications' performance and responsiveness under normal loads. In the future, it may be feasible to expand the functionality of the pipeline, including support for new types of visualization and EEG preprocessing methods.

The conducted study verified the applicability of an ontology-driven solution for mobile EEG visualization task on a new device, and also allowed us to expand the toolset of our platform with new visualization operator.

Acknowledgments

The authors express their deep gratitude to the Educational and Scientific Laboratory of Sociocognitive and Computer Linguistics, Faculty of Philology, Perm State University, for the provided equipment and support for the research.

References

1. K. E. Mathewson, J. P. Kuziek, J. E. M. Scanlon, D. Robles, The moving wave: Applications of the mobile EEG approach to study human attention, Psychophysiology 61 (2024). doi:10.1111/psyp.14603.

2. J. Sabio, N. S. Williams, G. M. McArthur, N. A. Badcock, A scoping review on the use of consumer-grade EEG devices for research, PLOS ONE 19 (2024) e0291186. doi:10.1371/ journal.pone.0291186.

3. K. Ryabinin, S. Chuprina, I. Labutin, Ontology-driven tools for EEG-based neurophysiological research automation, Scientific Visualization 13 (2021). doi:10.26583/sv.13.4.08.

4. A. Stopczynski, C. Stahlhut, J. E. Larsen, M. K. Petersen, L. K. Hansen, The smartphone brain scanner: A portable real-time neuroimaging system, PLoS ONE 9 (2014) e86733. doi:10.1371/journal.pone.0086733.

5. E. D. McKenzie, A. S. P. Lim, E. C. W. Leung, A. J. Cole, A. D. Lam, A. Eloyan, D. K. Nirola, L. Tshering, R. Thibert, R. Z. Garcia, E. Bui, S. Deki, L. Lee, S. J. Clark, J. M. Cohen, J. Mantia, K. T. Brizzi, T. R. Sorets, S. Wahlster, M. Borzello, A. Stopczynski, S. S. Cash, F. J. Mateen, Validation of a smartphone-based eeg among people with epilepsy: A prospective study, Scientific Reports 7 (2017). doi:10.1038/srep45567.

6. P. Samimisabet, L. Krieger, T. Nethar, G. Pipa, Introducing a new mobile electroencephalography system and evaluating its quality in comparison to clinical electroencephalography, Sensors 23 (2023) 7440. doi:10.3390/s23177440.

7. S. Blum, S. Debener, R. Emkes, N. Volkening, S. Fudickar, M. G. Bleichner, Eeg recording and online signal processing on android: A multiapp framework for braincomputer interfaces on smartphone, BioMed Research International 2017 (2017) 1–12. doi:10.1155/2017/3072870.

8. A. Campbell, T. Choudhury, S. Hu, H. Lu, M. K. Mukerjee, M. Rabbi, R. D. Raizada, Neurophone: brain-mobile phone interface using a wireless eeg headset, in: Proceedings of the second ACM SIGCOMM workshop on Networking, systems, and applications on mobile handhelds, SIGCOMM '10, ACM, 2010, pp. 3–8. doi:10.1145/1851322.1851326.

9. G. M. Rojas, J. A. Fuentes, M. Gálvez, Mobile device applications for the visualization of functional connectivity networks and eeg electrodes: ibrain and ibraineeg, Frontiers in Neuroinformatics 10 (2016). doi:10.3389/fninf.2016.00040.

10. G. Frishkoff, P. LePendu, R. Frank, H. Liu, D. Dou, Development of neural electromagnetic ontologies (nemo): Ontology-based tools for representation and integration of event-related brain potentials, Nature Precedings (2009). doi:10.1038/npre.2009.3458.1.

11. S. J. R. Méndez, Modeling actuations in bci-o: a context-based integration of sosa and iot-o, in: Proceedings of the 8th International Conference on the Internet of Things, IOT '18, ACM, 2018, pp. 1–6. doi:10.1145/3277593.3277914.

12. S. Chuprina, I. Labutin, A high-level adaptation toolkit for unified integration of software systems with neural interfaces, Scientific Visualization 16 (2024) 11–24. doi:10.26583/ sv.16.4.02.

13. A. Stopczynski, C. Stahlhut, M. K. Petersen, J. E. Larsen, C. F. Jensen, M. G. Ivanova, T. S. Andersen, L. K. Hansen, Smartphones as pocketable labs: Visions for mobile brain imaging and neurofeedback, International Journal of Psychophysiology 91 (2014) 54–66. doi:10.1016/j.ijpsycho.2013.08.007.

14. I. A. Labutin, S. I. Chuprina, Ontology-based neurointerface iot integration approach, Proceedings of the Institute for System Programming of the RAS 36 (2024) 91–108. doi:10.15514/ispras-2024-36(2)-8.

15. K. Ryabinin, S. Chuprina, I. Labutin, Tackling IoT Interoperability Problems with OntologyDriven Smart Approach, Springer International Publishing, 2021, pp. 77–91. doi:10.1007/ 978-3-030-89477-1_9.

16. S. I. Chuprina, I. A. Labutin, Scientific visualization tools to improve utilizing neural interface, in: Proceedings of the 32nd International Conference on Computer Graphics and Vision, GraphiCon-2022, Keldysh Institute of Applied Mathematics, 2022, pp. 391–402. doi:10.20948/graphicon-2022-391-402.

17. K. Ryabinin, S. Chuprina, Development of ontology-based multiplatform adaptive scientific visualization system, Journal of Computational Science 10 (2015) 370–381. URL: http://www.sciencedirect.com/science/article/pii/S1877750315000307. doi:https://doi.org/10.1016/j.jocs.2015.03.003.

18. S. I. Chuprina, Using data fabric architecture to create personalized visual analytics systems in the field of digital medicine, Scientific Visualization 15(5) (2023) 50–63. URL: https://api.semanticscholar.org/CorpusID:266356696. doi:10.26583/sv.15.5.05.

19. S. I. Chuprina, Adaptatsiya tekhnologii fabrik dannykh k razrabotke sistem vizual’noi analitiki v oblasti tsifrovoi meditsiny, in: Trudy 33 Mezhdunarodnoi konferentsii po komp’yuternoi grafike i mashinnomu zreniyu "GrafiKon-2023", Institut problem upravleniya im. V.A. Trapeznikova RAN, 2023, pp. 405–416. doi:10.20948/ graphicon-2023-405-416.