2018 ACM/IEEE 40th International Conference on Software Engineering: New Ideas and Emerging Results

Understanding the Impact of Pair Programming on the Minds of Developers

Sara Busechian
Department of Veterinary Medicine,
University of Perugia
via San Costanzo 4
Perugia, Italia I-06126
sarabusechian@gmail.com

Vladimir Ivanov
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
v.ivanov@innopolis.ru

Alan Rogers
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
a.rogers@innopolis.ru

Ilyas Sirazitdinov
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
i.sirazitdinov@innopolis.ru

Giancarlo Succi
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
g.succi@innopolis.ru

Alexander Tormasov
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
tor@innopolis.ru

Jooyong Yi
Innopolis University
Universitetskaya St, 1
Innopolis, Respublika Tatarstan,
Russia 420500
j.yi@innopolis.ru

ABSTRACT

Software is mostly, if not entirely, a knowledge artifact. Software best practices are often thought to work because they induce more productive behaviour in software developers. In this paper we deployed a new generation tool, portable multichannel EEG, to obtain direct physical insight into the mental processes of working software developers engaged in their standard activities. We have demonstrated the feasibility of this approach and obtained a glimpse of its potential power to distinguish physical brain activity of developers working with different methodologies.

KEYWORDS

Brain sciences, empirical studies, pair programming.

ACM Reference format:
Sara Busechian, Vladimir Ivanov, Alan Rogers, Ilyas Sirazitdinov, Giancarlo Succi, Alexander Tormasov, and Jooyong Yi. 2018. Understanding the Impact of Pair Programming on the Minds of Developers. In *Proceedings of 40th International Conference on Software Engineering: New Ideas and Emerging Results Track, Gothenburg, Sweden, May 27-June 3, 2018 (ICSE-NIER'18),* 4 pages.
https://doi.org/10.1145/3183399.3183413

ICSE-NIER'18, May 27-June 3, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5662-6/18/05…$15.00
https://doi.org/10.1145/3183399.3183413

1 INTRODUCTION

Software development is the result of our ability to elaborate ideas into algorithms, and is therefore mostly the direct result of mental activities. As a consequence, software best practices must work because in some fashion they induce more productive behaviour in software developers. Obviously it is of great interest to understand more and more directly how suggested practices affect mental activity to promote productivity. Still, despite its industrial relevance it has not yet investigated in depth how such behaviours lead to greater productivity [8].

Our **new idea** is to use the new generation of wearable multichannel EEG devices to understand in depth the mental processes occurring in software engineers and developers during the different phases of development, and how these processes may differ with the use of different development approaches. Such a deep understanding should enable us to detect the most effective conditions, processes, and practices to develop high quality software systems with optimum efficiency.

One of the most studied novel software development techniques is pair programming. Some studies have suggested that pair programming yields productivity comparable or even greater than that achieved by two individual programmers working separately like work of di Bella et al. (2013) [3]. In spite of these observations, the practice is still widely rejected as “obviously" less efficient, perhaps in large part because there is no clear mechanism by which pairing could be explained to yield productivity bonuses. Hypotheses have been made; an example of such hypotheses is that pair programming induces higher levels of concentration [17], thus reducing the amount of defects inserted in the code. However, it is not clear whether this is indeed true, since the evidence around such claims is very indirect and circumstantial – in the work above, it is based on analysis of the frequency of switching between terminal windows. We provide evidence for the applicability of this idea to an evaluation of the brain activity of programmers with a preliminary examination comparing such activity while engaged in pair programming and solo programming.

Some work has been done applying emerging techniques for examination of brain activity in programmers, as will be discuss later in Section 2. That effort, however, was not representative of real development environments, cannot feasibly be extended to such environments and therefore to real activities, and does not permit the use of reliable or accurate data collection devices.

In Section 3 we detail our new idea. In Section 4 we perform initial experimentation of our new idea, to gain an early assessment of its viability for this application. In Section 5 we evaluate the results of the experimentation. In Section 6 we discuss our overall findings. Finally, in Section 7 we draw some conclusions and we outline how we would like to carry our research forward.

2 BACKGROUND

Other studies have recently tried to understand the work of the mind during software development with various purposes.

Full usage of the EEG. A complete portable EEG device has been used in software engineering only by Lee et al. (2016) [11] to explore the differences in mind activities between novice and experts in program comprehension tasks, with research goals similar to the ones of [7] but a completely different from ours.

Functional magnetic resonance imaging. Functional magnetic resonance imaging (fMRI) provides indirect estimates of brain activity, measuring metabolic changes in blood flow and oxygen consumption as a result of increased underlying neural activity. This technique allows the detection of active regions of the brain [4]. As a result fMRI is widely used to determine specific brain regions which are responsible for certain mental activity. In order to learn about software developers' brain activity, researchers chose code review and code comprehension as the primary activities for which brain activity needs to be understood [4, 15, 16]. Siegmund et al. (2014) detected activation specific Broadmann-areas during code comprehension [15]. In their followup work (2017) they investigated the difference between bottom-up program comprehension and comprehension with semantic cues in terms of brain areas involved [16]. This study uses very accurate techniques to explore the work of the brain, the fMRI. However, the purposes of such studies are quite different to this one, since they aim mostly to explore the mental processes rather to identify what makes software processes more efficient and effective. Moreover, the conditions of such empirical evaluations cannot in any way compare to those of the standard software development environments - their subjects were involved in sessions of 30 minutes lying on a medical device and looking at pieces of code in a mirror [16].

Ensemble of bio-metric sensors. An alternative approach has been to use an ensemble of bio-metric sensors like eye trackers for measuring pupil size and eye blinks, electroencephalography to determine brain activity, electro-dermal activity sensors to detect skin-related activity, and heart-related sensors [6, 12, 19].

This approach was applied in a series of investigations which will be described below. The main interest in these investigations was to obtain metrics that correlate with software developers performance. Züger and Fritz (2015) used interruptibility [19] while Müller and Fritz (2015) used positive and negative emotions of software developers [12] as metrics of progress in the change task. They processed the data from multiple bio-sensors and applied methods of supervised learning (Naive Bayes) to distinguish levels of these cognitive states [5, 12, 19].

The problem of this approach is that the devices using to collect the data were mostly focused on collecting emotions and the data analysis was focused on finding correlation between emotions and progress, which was the core of the study. Monitoring the state of the mind in depth was not their purpose so that analysis was not precise, and was also limited because:

- the assessment of emotions was performed subjectively by the participants [12];
- a single channel EEG device was used, which may result in an error of up to 50% [13].

3 OUR NEW IDEA

We propose to use a wireless full channel EEG device to measure brain activity of software developers. Electroencephalography (EEG) was widely used in the tasks of measuring different emotional and cognitive states [1]. Subsequently EEG found an application in measuring performance metrics such as task engagement and mental workload [2]. However, there is a lack of studies of software developers' brain activity using pure EEG signal, apart from the previously mentioned study of Lee et al. (2016) [11] aimed at comparing novices and experts.

3.1 Measurement of the brain activity

As mentioned, the fMRI data acquisition approach differs from ours in that the fMRI approach restricts the subjects of the experimentation to working in a lying position with the ability to see only a few lines of code via a mirror [15, 16], while the EEG we adopted allows the subjects of the experimentation to work in their usual working configuration, performing their regular tasks.

In contrast to the 1-channel EEG device used in the works of Fritz and Müller (2015, 2016) [12, 19], the multichannel EEG data collection has two substantial advantages:

- The usage of multichannel EEG allows us to apply advanced signal decomposition methods like independent component analysis (ICA). Through ICA it is possible to apply the most effective algorithms for the removal of artifacts in the EEG, like EOG and muscle activity [9].
- The multichannel EEG places electrodes all over the head. Therefore, it allows us to obtain a brain map with different brain areas' activity. Consequently it is possible to detect the most "active" parts of the brain and this activity may be associated with some specific performance metrics.

3.2 Proposed measurement device

In our research we used a 24-channel wireless EEG Smart BCI cap which was provided by Mitsar Company. The placement of electrodes is according to standard 10-20 scheme. The frequency band is 0 - 70 Hz, sampling rate is 250 Hz, input range is 300 mV. EEG was recorded at 20 scalp sites (F3 and etc.). EEG Studio by Mitsar Company was used to record data, to preprocess it and to perform frequency analysis. The device does not cause any particular impediment to developers, as it also appeared from a suitable survey among the participants of the pilot study on pair programming.

4 FIRST EXPERIMENTATION OF THE NEW IDEA

To verify the suitability of our approach we have experimented it in a simple pair programming session. Our goal was to see if earlier work hypothesizing that pair programming induces higher level of concentration would find support from the analysis of brain waves.

The experimentation was organized in four phases:

(1) First, the EEG device was calibrated, recording the brain waves during a relaxed state with closed eyes and a relaxed state with opened eyes for 5 minutes each.
(2) Second, each participant solved his real task in a regular solo mode for 40 minutes.
(3) Third, after a 10 minutes break there was a 40 minutes session in which the participants worked on their task in pair programming in "driver" mode.
(4) Fourth, after a 10 minutes break, the "driver" switched his position to the "navigator" and the pair continued to work on their problem for a further 40 minutes.

The EEG cap was not removed during the breaks to maintain the same electrode placement and impedance.

The subjects of our experimentation were 4 male computer science students of around 23 years of age who were enrolled in a computer science graduate program in Russia. Each participant had more than 3 years experience in programming. Their technological stack during the experiment consisted of JavaScript, React JS, Python and Scala.

The participants were instructed beforehand to take all typical measures for these kinds of activities, e.g., to wash and dry their heads, to avoid touching the electrodes, etc. As mentioned at the end of the experimentation, the participants were asked whether the EEG devices caused discomfort or triggered any unusual action, and the answers were negative to both questions. The overall experimentation was run twice.

5 EVALUATION OF THE RESULTS OBTAINED FROM THE FIRST EXPERIMENTATION

First of all, we would like to emphasize that the goal of the experimentation is primarily to answer the question of the suitability of the approach. Based on the preliminary experience, the approach appears absolutely practical and effective. The required devices have a cost no higher than a professional workstation and the use of the devices does not cause any problems. Interpretation of the results does require specific expertise, and a member of the experimental team (and author of this paper) is an expert in neuroimaging and member of a medical school.

The results of the data collection are presented in Figure 1 for solo programming and in Figure 2 for pair programming for the same person. They contain the asymmetrical topographic brain

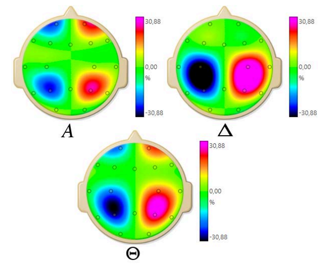

Figure 1: Asymmetrical topographic brain maps for Alpha (A), Delta (Δ), and Theta (Θ) waves during solo programming

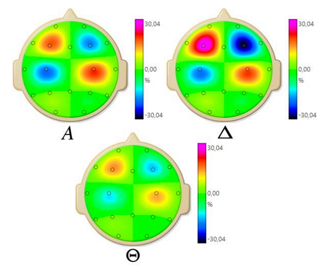

Figure 2: Asymmetrical topographic brain maps for Alpha (A), Delta (Δ), and Theta (Θ) waves during pair programming

maps for alpha, delta, and theta waves, which are typically used to understand the state of the brain [1, 2, 10, 14].

The first comment is that it is evident that solo and pair programming induce different brain waves: the two set of tomographic images are clearly very different.

We first consider the alpha waves, which are typical of rest conditions. Looking at the two pictures, we notice that in the case of pair programming the picture has in general more waves (changes of colors from deep blue to deep red) in the central and frontal part of the brain, while it is more green in the back. The alpha waves are typical of relaxed period. In particular, their presence in the occipital lobe reflect states of closed eyes, while their presence in the center or in the front indicates the presence of activities done during relaxation. This can be interpreted as the fact that during pair programming people are active in a relaxed mode, while during solo the relaxation occurs when people close their eyes, for instance to take a break, thus somehow confirming an earlier study [18] that pair programming induced also a less stressful pattern of work.

Delta waves indicate the presence of tasks requiring concentration to solve a task. We notice that they are significantly present in pair programming and in solo programming, with a slightly larger but less intense coverage in pair programming – however such small variations could be purely due to collateral situations.

Theta waves denote the presence of a desire to suppress actions (including thinking). We notice that they are much more present in the situation of solo programming than in pair programming. This could lead us to interpret it as the desire to eliminate stimulus dragging the attention away to stay concentrated on a task.

Altogether, a "reasonable" interpretation of this limited tomographic images could lead us to think that during pair programming developers are more relaxed than when doing solo programming, exercise their mind in a similar way to solve tasks, but do not need to put effort in repressing distracting stimulus. During solo programming the relaxation occurs by dragging the thought away from the tasks and there is the need of a specific coercion to be concentrated on the work. Indeed, this interpretation is well aligned with the mentioned previous work [17].

6 DISCUSSION

In this paper we are claiming that since software development involves a significant amount of mental activities, a full EEG can be successfully applied to evaluate the effectiveness of various software development processes and practices. Moreover, we claim that this is now becoming effectively doable, due to the availability of portable EEG devices. To this end we have carried out an early, practical experimentation to perform a preliminary study on the impact of pair programming on brain activity. A result of this early experimentation is that the approach is, indeed, doable, and that the outcome of the experimentation identifies a very different brain activity between pair programming and solo programming.

While we think that we have substantiated properly the effective applicability of our "new idea" to a wide number of empirical studies, our findings on pair programming should be subject to further experimentation in which we should:

- experiment with subjects of larger volume and more diversity;
- change the sequence of the activities of solo and pair programming to verify that it does not have an effect on the results;
- investigate further the interpretation of the resulting brainwaves.

7 CONCLUSION

To gain a greater understanding of how software developers' brains work is obviously of tremendous interest and value. The results we have shared suggest that advanced wearable multichannel EEG can clearly distinguish between the activity patterns of brains involved in different approaches to the superficially similar activity of creating software code. It has long been very clear that the results of this activity differ among individual practitioners and among approaches, but the only available tools to analyze the sources of these differences have been crude and indirect. As a result it has been difficult either to make clear, evidence-based recommendations or to explore in detail a variety of techniques.

The results we share provide only the most preliminary insight. Primarily they demonstrate the viability of this approach to offer much greater insights with additional work. Many of the extensions are obvious and give hope of providing quantitative, physical, insights into the mechanism of pair programming and other activities, like fixing, modifying, and maintaining code, forecasting effort and improving quality of systems under development, etc. In addition, we need to find suitable mechanisms to compare (possible automatically) several sets of the brain waves to be able to compare and generalize from greater numbers of experimental observations.

ACKNOWLEDGMENTS

We would like to thank Innopolis University for generously funding this research, and Medical Computer Systems and Mitsar Co for supplying us the essential devices for performing this study.

REFERENCES

[1] Soraia M. Alarcao and Manuel J. Fonseca. 2017. Emotions Recognition Using EEG Signals: A Survey. *IEEE Transactions on Affective Computing* 20, 99 (June 2017), 1–1.

[2] Chris Berka, Daniel J. Levendowski, Michelle N. Lumicao, Alan Yau, Gene Davis, Vladimir T. Zivkovic, Richard E. Olmstead, Patrice D. Tremoulet, and Patrick L. Craven. 2007. EEG correlates of task engagement and mental workload in vigilance, learning, and memory tasks. *Aviation, space and environmental medicine* 78, 5 (May 2007), 231–244.

[3] Enrico di Bella, Ilenia Fronza, Nattakarn Phaphoom, Alberto Sillitti, Giancarlo Succi, and Jelena Vlasenko. 2013. Pair Programming and Software Defects–A Large, Industrial Case Study. *IEEE Transactions on Software Engineering* 39, 7 (2013), 930–953. DOI:http://dx.doi.org/doi.ieeecomputersociety.org/10.1109/TSE.2012.68

[4] Benjamin Floyd, Tyler Santander, and Westley Weimer. 2017. Decoding the representation of code in the brain: an fMRI study of code review and expertise. In *Proceedings of the 39th International Conference on Software Engineering, ICSE 2017, Buenos Aires, Argentina, May 20-28, 2017.* 175–186.

[5] Thomas Fritz, Andrew Begel, Sebastian C. Müller, Serap Yigit-Elliott, and Manuela Züger. 2014. Using Psycho-physiological Measures to Assess Task Difficulty in Software Development. In *Proceedings of the 36th International Conference on Software Engineering (ICSE 2014).* ACM, New York, NY, USA, 402–413.

[6] Thomas Fritz and Sebastian C. Müller. 2016. Leveraging Biometric Data to Boost Software Developer Productivity. In *Leaders of Tomorrow Symposium: Future of Software Engineering, FOSE@SANER 2016, Osaka, Japan, March 14, 2016.* 66–77.

[7] Ilenia Fronza, Alberto Sillitti, and Giancarlo Succi. 2009. An Interpretation of the Results of the Analysis of Pair Programming During Novices Integration in a Team. In *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM '09).* IEEE Computer Society, 225–235. DOI:http://dx.doi.org/10.1109/ESEM.2009.5315998

[8] Vladimir Ivanov, Alan Rogers, Giancarlo Succi, Jooyong Yi, and Vasily Zorin. 2017. What Do Software Engineers Care About? Gaps Between Research And Practice. In *Proceedings of the 2017 ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC/FSE 2017).*

[9] Uriguen JA and Garcia-Zapirain B. 2015. EEG artifact removal-state-of-the-art and guidelines. *Journal of neural engineering* 78, 12 (April 2015), 1–23.

[10] Elif Kirmizi-Alsan, Zubeyir Bayraktaroglu, Hakan Gurvit, Yasemin H. Keskin, Murat Emre, and Tamer Demiralp. 2006. Comparative analysis of event-related potentials during Go/NoGo and CPT: Decomposition of electrophysiological markers of response inhibition and sustained attention. *Brain Research* 1104, 1 (2006), 114 – 128.

[11] SeolHwa Lee, Andrew Matteson, Danial Hooshyar, SongHyun Kim, JaeBum Jung, GiChun Nam, and HeuiSeok Lim. 2016. Comparing Programming Language Comprehension between Novice and Expert Programmers Using EEG Analysis. In *16th IEEE International Conference on Bioinformatics and Bioengineering, BIBE 2016, Taichung, Taiwan, October 31 - November 2, 2016.* 350–355.

[12] Sebastian C. Müller and Thomas Fritz. 2015. Stuck and Frustrated or in Flow and Happy: Sensing Developers' Emotions and Progress. In *37th IEEE/ACM International Conference on Software Engineering, ICSE 2015, Florence, Italy, May 16-24, 2015, Volume 1.* 688–699.

[13] Ignas Martisius Rytis Maskeliunas, Robertas Damasevicius and Mindaugas Vasiljevas. 2016. Consumer-grade EEG devices: are they usable for control tasks? *PeerJ* 4 (March 2016), 1–22.

[14] Donald L. Schomer and Fernando L. da Silva. 2012. *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields.* Wolters Kluwer Health.

[15] Janet Siegmund, Christian Kästner, Sven Apel, Chris Parnin, Anja Bethmann, Thomas Leich, Gunter Saake, and André Brechmann. 2014. Understanding understanding source code with functional magnetic resonance imaging. In *36th International Conference on Software Engineering, ICSE '14, Hyderabad, India - May 31 - June 07, 2014.* 378–389.

[16] Janet Siegmund, Norman Peitek, Chris Parnin, Sven Apel, Johannes Hofmeister, Christian Kästner, Andrew Begel, Anja Bethmann, and André Brechmann. 2017. Measuring neural efficiency of program comprehension. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ESEC/FSE 2017, Paderborn, Germany, September 4-8, 2017.* 140–150.

[17] Alberto Sillitti, Giancarlo Succi, and Jelena Vlasenko. 2012. Understanding the Impact of Pair Programming on Developers Attention: A Case Study on a Large Industrial Experimentation. In *Proceedings of the 34th International Conference on Software Engineering (ICSE '12).* IEEE Press, Piscataway, NJ, USA, 1094–1101.

[18] Giancarlo Succi, Witold Pedrycz, Michele Marchesi, and Laurie Williams. 2002. Preliminary analysis of the effects of pair programming on job satisfaction. In *Proceedings of the 3rd International Conference on Extreme Programming (XP).* 212–215.

[19] Manuela Züger and Thomas Fritz. 2015. Interruptibility of Software Developers and its Prediction Using Psycho-Physiological Sensors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015.* 2981–2990.