

KÜLÖNBÖZŐ KÓDHIBA-KERESÉSI MEGKÖZELÍTÉSEK HATÉKONYSÁGÁNAK VIZSGÁLATA SZEMKÖVETŐ RENDSZERREL

EXAMINING THE EFFECTIVENESS OF DIFFERENT DEBUGGING APPROACHES WITH AN EYE TRACKING SYSTEM

Katona Jozsef^{0000-0002-8188-2425 1,2*}

¹ Dunaújvárosi Egyetem, Informatikai Intézet, Szoftverfejlesztési és Alkalmazási Tanszék, Dunaújváros, Magyarország

² Óbudai Egyetem, Kandó Kálmán Villamosmérnöki Kar, Elektronikai és Kommunikációs Rendszerek Intézet, Műszertechnikai és Automatizálási Tanszék, Budapest, Magyarország

<https://doi.org/10.47833/2023.1.CSC.003>

Kulcsszavak:

programming
eye-tracking
debugging

Keywords:

programozás
szemmozgás követés
hibakeresés

Cikktörténet:

Beérkezett 2022. október 10.
Átdolgozva 2022. október 31.
Elfogadva 2022. november 5.

Összefoglalás

A tanulmány a szemmozgáskövető rendszerek alkalmazhatóságát vizsgálja egy programozási feladat tekintetében, amely során egy helytelenül működő algoritmus hibáinak feltárása és kijavítása alatt megfigyelésre, rögzítésre, valamint kiértékelésre kerültek a szemmozgás paraméterek. A vizsgálat során néhány, a tesztalanyokra jellemző paraméter alkalmazásával, két csoport került kialakításra, ahol az első csoport hibakeresés során inkább az apróbb módosításokat és a gyakoribb fordítások és futtatások technikáját alkalmazta, amíg a második csoport tagjai nagyobb hangsúlyt fektettek az értelmezésre. A kiértékelés során a két csoport szemmozgás követésére jellemző paraméterek, valamint ezen csoportok hatékonysági mutatói kerültek elemzésre statisztikai próbák alkalmazásával. Az eredmények azt mutatták, hogy a kevesebb kapkodás, az alaposabb megfontoltság és figyelem esetében kevesebb információmennyiség feldolgozása is elegendő a hatékonyabb hibakereséshez.

Abstract

The study examines the suitability of eye movement tracking systems for a programming task, during which the parameters of eye movement were monitored, recorded and evaluated in order to discover and correct the errors of a malfunctioning algorithm. During the investigation, based on some parameters typical of the test subjects, two groups were formed. During the debugging of the first group, smaller modifications and the technique of more frequent translations and runs were used, while the members of the second group placed more emphasis on interpretation. During the evaluation, the eye movement tracking parameters of the two groups, as well as the efficiency

* Kapcsolattartó szerző.
E-mail cím: katonaj@uniduna.hu

indicators of these groups, were analyzed using statistical tests. The results show that processing less amount of information is sufficient for fault-finding in the case of less haste, more thorough consideration and attention.

1. Bevezetés

A szoftverfejlesztés mostanra kritikusan bonyolult és összetett folyamattá vált, mivel az implementálandó, majd később továbbfejlesztendő és karbantartandó forráskódok előállítására egyre nehezebb feladat. A folyamat megkönnyítése érdekében a fejlesztők egyre újabb technikákkal, technológiákkal [1], szimulációval [2], VR-alapú megoldásokkal [3], módszerekkel [4], fejlesztésekkel [5] és tanulási technikákkal [6] állnak elő. Azonban az ipari és oktatási visszajelzések alapján a megfelelő eszköztár kiválasztása nem minden esetben egyértelmű feladat, többek között az egyénekre jellemző, eltérő kognitív képességek miatt [7-9]. A programozás, mint összetett kognitív tevékenység objektív vizsgálata hozzájárulhat az eredményesebb alkalmazások és szoftverrendszerek implementációjához [10], megvalósításához, valamint a fejlesztőkhöz legjobban illeszkedő eszköztár kiválasztásához [11]. Az ember-számítógép interfészek csoportjába sorolható szemmozgáskövető rendszerek lehetővé teszik, hogy az emberi tekintet útvonalelemzésével objektív módon vizsgálható legyen egy szoftver forráskódjának olvashatósága, érthetősége, összességében a kognitív terhelés.

A közelmúltban egyre több olyan kutatás jelent meg, amelyek a komplexebb mentális tevékenységet igénylő szoftverfejlesztési fázisok költségeinek csökkentését célozták meg szemmozgáskövető rendszerek alkalmazásával. Emellett a hiteles adatkezelés is fontos jelentőséggel bír az adatgyűjtés során [12].

A kutatások egy csoportja a fejlesztési ciklusok közül a tervezési fázist vizsgálja. A csoporton belül egyesek az objektum-orientált programok számára keretet biztosító UML osztálydiagramok tervezéséhez alkalmazott vizuális minták, valamint azok alkalmazásának kognitív hatását [13-15], amíg mások ezen diagramok elrendezését és az ebből fakadó hatékonyabb érthetőséget és olvashatóságot elemzik [16-18].

A kutatások egy másik csoportja az implementációs tevékenység produktumát, azaz a létrehozott forráskódok olvashatóságát, érthetőségét és összetettségét elemzik [19,20]. Crosby és Stelovsky egy tanulmányában azt elemezte, hogy a vizsgálati személyek milyen módon olvasnak egy forráskódot, illetve anyanyelvű szöveget. A kutatás eredményeként megfogalmazták, hogy szignifikáns különbség mutatható ki a két szövegtípus olvasási és áttekintési módjában, ugyanis a programkódok esetében nagyobb rögzítési idők voltak jelen. [21] Crosby, Schlotz és Wiedenbeck megfigyelték, hogy a kezdő programozók sokkal nagyobb hangsúlyt fektetnek a kommentek, kiegészítő információk olvasásával, mint a tapasztaltabb társaik [22].

A cikk célja, hogy különböző hibafeltárási és javítási megközelítéseket hasonlítsa össze teszt alanyok bevonásával és szemmozgás paraméterek kiértékelésével. A 2. fejezet ismerteti a vizsgálati személyeket, a kutatáshoz felhasznált eszközöket, továbbá a kutatás lefolytatásának lépéseit és körülményeit, amíg a 3. fejezetben olvasható többek között a felhasznált informatikai rendszer által rögzített eredmények kiértékelése és a meghatározott eredmények által megfogalmazható következtetések. A 4. fejezetben az összefoglalás és konklúzió olvasható.

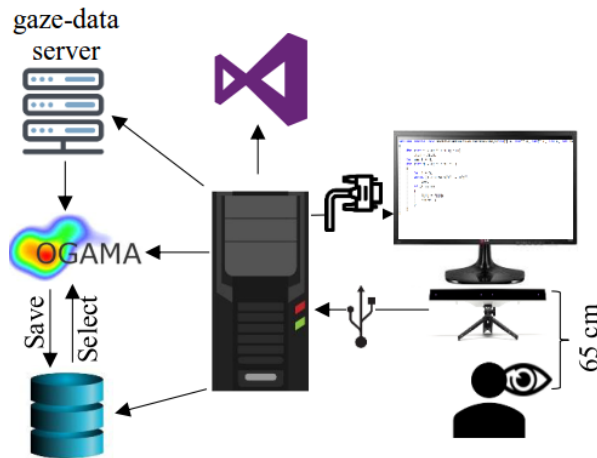
A témához szorosan kapcsolódó feldolgozott irodalmak, valamint a kutatási célokban megfogalmazottak alapján az alábbi kutatási kérdés került meghatározásra:

Kimutatható-e összefüggés a teszt alanyok szemmozgás paramétereiben, valamint a helytelenül működő algoritmus hibakeresési és javítási megközelítések között?

2. A vizsgálat elvégzésénél alkalmazott eszközök és a vizsgálat lefolytatásának körülményei

Az alkalmazott szemmozgás megfigyelő és rögzítő informatikai rendszerben többek között egy GazePoint 3 (GP3) hardveregység (<https://www.gazept.com/product/gazepoint-gp3-eye-tracker/>), valamint a nyíltforráskódú és ingyenesen elérhető OGAMA szoftvercsomag (<http://www.ogama.net/>) került felhasználásra. Ezeket az eszközöket korábban már több kutatás során is eredményesen

alkalmazták [23-26]. A rendszer további részének tekinthető egy LG 22M45 típusjelzésű 1920x1080 felbontásra képes 22" átmérőjű monitor, valamint a Visual Studio, amelyek segítségével a módosított algoritmus került megjelenítésre. Az összeállított informatikai rendszer sematikus nézete az 1. ábrán látható.



1. ábra. Az összeállított informatika teszrendszer sematikus nézete [27].

2.1. Tesztalanyok

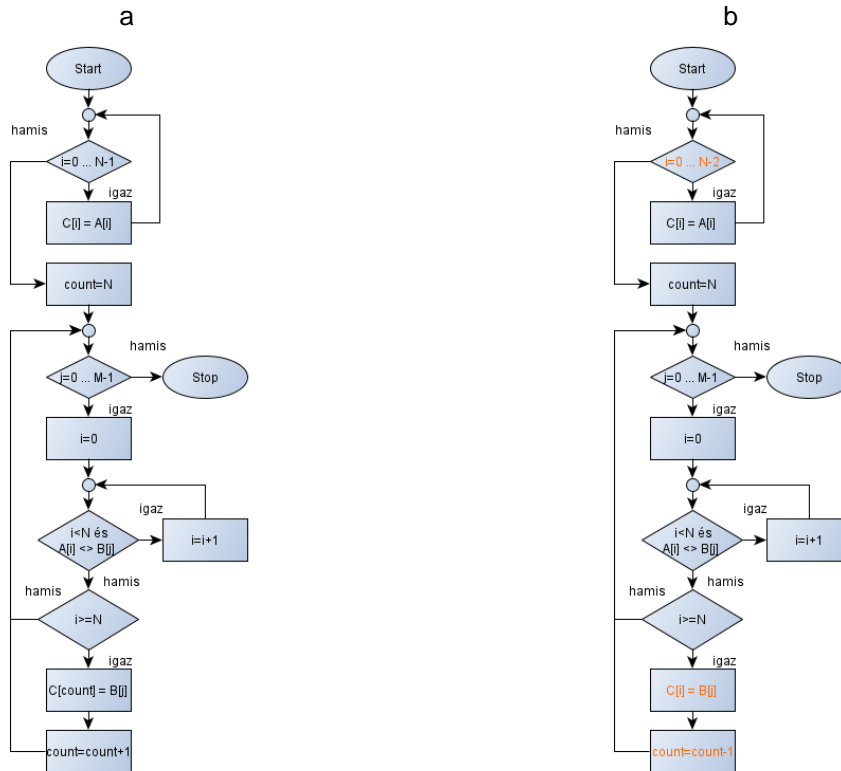
A kutatásban összesen 22 személy (10 nő és 12 férfi) került bevonásra, akik magukat teljesen egészségesnek vallották és semmilyen gyógyszert nem szedtek. A 18 és 22 év ($M=19,76$ $SD=1,31$) közötti egyetemi hallgatók önkéntes alapon jelentkeztek abban az esetben, ha a Bevezetés a Programozásba című tantárgyat már korábban sikeresen teljesítették, mivel ez a tantárgy tartalmazza a teszthez szükséges algoritmus ismereteket.

2.2. A felhasznált vizsgálati algoritmus

A hibafeltáráshoz és javításhoz egy olyan algoritmus (tömb egyesítés, vagy más néven unióképzés) került kiválasztásra (2.a ábra), amelyet a teszt alanyok korábbi tanulmányaikból már jól ismertek. Az algoritmus során N és M elemű halmazokat (azokat az elemeket, amelyek legalább az egyik halmazban jelen vannak) egyesítünk egy C vektorba. Az 2.a ábrán látható, helyesen működő algoritmusban három darab hiba került elrejtésre:

1. A *for* ciklus feltételes utasítása módosítva lett $N-1$ -ről $N-2$ -re, ennek eredményként a C vektorba az A vektor utolsó eleme nem kerülhet.
2. Az *if* elágazáson belül a C vektor index hivatkozása *count*-ról i -re változott, ennek következménye, hogy a C vektor N -ik ($A.length+1$) pozícióba kizárólag a B vektor azon utolsó eleme fog kerülni, amely nem található meg az A vektorban is.
3. Ugyanebben a feltételes elágazásban az inkrementáló kifejezés ($count(count+1)$) dekrementálóra változott ($count(count-1)$), így C vektor N -ik ($A.length+1$) pozícióba kizárólag a B vektor azon első eleme fog kerülni, amely nem található meg az A vektorban is.

A módosított egyesítés algoritmus folyamatábrája látható az 2.b ábrán (a módosítások piros betűszínnel kerültek kiemelésre). A lentebb bemutatott és módosított algoritmus a tesztalanyok számára leginkább ismert Visual Studio Community fejlesztő környezetben került implementálásra C# magasszintű programozási nyelven.



2. ábra. A helyes (a), illetve a helytelen (b) unióképzés algoritmusának leírása folyamatábrára segítségével.

2.3. A vizsgálat lefolytatásának körülményei és lépései

A vizsgálat a fentebb említett és bemutatott egységek, csomagok és algoritmusok alkalmazásával történt. A vizsgálat megkezdése előtt a szükséges szoftvercsomagok (pl.: Gazepoint Control, gaze-date szerver, OGAMA stb.) telepítésre és konfigurálásra, majd megnyitásra kerültek. Az GP3 eszköz a sikeres csatlakoztatást követően a monitor alá került elhelyezésre, a szemtől 65cm-es távolságra. A teszt alanyok arcát közvetlen napfény nem zavarhatta. A megfelelő elhelyezést követően a Visual Studio-ban prezentálásra került a módosított algoritmus. A teszt alanyok részletes tájékoztatást kaptak a tesztelés menetéről. Az eszköz sikeres kalibrációját követően elkezdhető volt a hibák keresése és javítása. A célidő 3 percben került meghatározásra, ez idő alatt mindenki annyiszor fordíthatta és futtathatta az algoritmus ahányszor csak akarta, az idő letelte után azonban abba kellett hagyni a munkát. A folyamat során szemmozgási paraméterek rögzítésre, majd a teszt végeztével mentésre kerültek a további statisztikai kiértékelés céljából.

3. Eredmények és következtetések

Az eredmények kiértékelése során – a probléma megoldás módszere alapján – két csoport került meghatározásra, továbbá nem csak a forráskódra vonatkozó kiértékelés, hanem a folyamatábrán történő hibakeresésre is kiterjedt.

3.1. Szemmozgás paraméterek kiértékelése

A változók értékeinek eloszlása mindkét csoport esetében Shapiro-Wilk (W) tesztet alkalmazva került meghatározásra, továbbá az adatok intervallum skálán mértek és függetlenek egymástól.

Az első szemmozgás mutatószám kiértékelése a fixációk mennyisége tekintetében történt és a forráskódra vonatkozott. Mivel a $W(15)=0,904$, $p=0,110$, illetve a $W(7)=0,901$, $p=0,339$, valamint ($F=0,021$, $p=0,887$), ezért kétmintás t-próba került alkalmazásra, amelynek eredménye ($t(20)=2,507$, $p=0,021$, $d=1,106$) alapján kijelenthető, hogy a két csoport fixációs darabszámának átlaga

szignifikánsan különbözik egymástól. A *Try&R* esetében $M=210,67$, $SD=50,06$ darab, amíg a *Think&R* estében $M=149,57$, $SD=59,98$ darab volt az átlagos fixációs mennyiség. [27]

A második szemmozgás mutatószám kiértékelése az átlagos fixációs időtartam tekintetében történt és a forráskódra vonatkozott. Mivel a $W(15)=0,960$, $p=0,691$, illetve a $W(7)=0,944$, $p=0,672$, valamint ($F=0,654$, $p=0,428$), ezért kétmintás t-próba került alkalmazásra, amelynek eredménye ($t(20)=-0,544$, $p=0,592$, $d=0,26$) alapján kijelenthető, hogy a két csoport fixációs időtartamának átlaga szignifikánsan nem különbözik egymástól. A *Try&R* esetében $M=364,01$, $SD=85,62$ milliszekundum, amíg a *Think&R* estében $M=383,96$, $SD=65,71$ milliszekundum volt az átlagos fixációs időtartam. [27]

A fentebbi, kizárólag a forráskódra kiterjedő szemmozgás paraméterek eredményeinek kiértékelését követően a folyamatábrán elvégzett hibakeresés során rögzített ugyanezen paraméterek kiértékelése is megtörtént.

Az első szemmozgás mutatószám kiértékelése a folyamatábrára vonatkozóan is a fixációk mennyisége tekintetében történt. Mivel a $W(15)=0,893$, $p=0,289$, illetve a $W(7)=0,899$, $p=0,093$, valamint ($F=0,042$, $p=0,839$), ezért kétmintás t-próba került alkalmazásra, amelynek eredménye ($t(20)=2,425$, $p=0,025$, $d=0,982$) alapján kijelenthető, hogy a két csoport fixációs darabszámának átlaga szignifikánsan különbözik egymástól. A *Try&R* esetében $M=187,73$, $SD=50,39$ darab, amíg a *Think&R* estében $M=128,14$, $SD=60,66$ darab volt az átlagos fixációs mennyiség.

A második szemmozgás mutatószám kiértékelése a folyamatábrára vonatkozóan is az átlagos fixációs időtartam tekintetében történt. Mivel a $W(15)=0,973$, $p=0,904$, illetve a $W(7)=0,924$, $p=0,504$, valamint ($F=0,205$, $p=0,656$), ezért kétmintás t-próba került alkalmazásra, amelynek eredménye ($t(20)=0,172$, $p=0,865$, $d=0,09$) alapján kijelenthető, hogy a két csoport fixációs időtartamának átlaga szignifikánsan nem különbözik egymástól. A *Try&R* esetében $M=305,6$, $SD=76,09$ milliszekundum, amíg a *Think&R* estében $M=312,39$, $SD=90,04$ milliszekundum volt az átlagos fixációs időtartam.

Az eredmények vizsgálatát követően megállapítható, hogy a *Try&R* tagjai esetében a sok kisebb módosítás, valamint a gyakoribb fordítás és futtatás kisebb hatékonyságot és nagyobb idő szükségletet mutat a forráskódban elrejtett hibák felderítésében és javításában, mint a *Think&R* tesztalanyainak esetében, amelynek oka feltételezhetően a bizonytalanság, a szerte ágazóbb figyelem, valamint a megfontoltság hiánya lehet.

A szemmozgás mutatószámainak kiértékelését követően kijelenthető, hogy a két csoport tagjai között szignifikáns eltérés csak a fixációk számában volt kimutatható. A *Try&R* tagjai több alkalommal végeztek fixálást a forráskódon, mint a *Think&R* tagjai, tehát a képernyő több pontjáról volt szükségük információfelvételre, azonban a fixációk időtartamában szignifikáns különbség nem volt kimutatható, amely azt jelentheti, hogy a *Try&R* alanyai ugyan több pontot vizsgáltak a forráskódon, azokat nem tanulmányozták alaposabban, amíg a *Think&R* esetében ezek az eredmények azt jelenthetik, hogy kisebb fixáció mennyiség mellett sem volt szükség hosszabb vagy rövidebb fixációs időtartamra, tehát a *Think&R* jobb eredménye leginkább annak köszönhető, hogy szignifikánsan kevesebb fixációra volt szükségük ahhoz, hogy a hibákat megtalálják és kijavítsák. Mivel az információ feldolgozási időtartamban nem mutatható ki szignifikáns eltérés, valamint a teszt alanyok mindegyike sikeresen abszolválta a Bevezetés a Programozásba című tárgyat, ezért a különbség nem magyarázható tudásbéli különbséggel. A tudás hiány feltehetően a nagyobb fixáció szám mellett nagyobb információ feldolgozási időtartamot. Azonban a kutatás egyik korlátjának tekinthető a teszt alany darab száma, így az eredmények ennek tekintetében veendő figyelembe. Összességében a kevesebb kapkodás, az alaposabb megfontoltság és figyelem esetében kevesebb információmennyiség feldolgozása is elegendő a hatékonyabb hibakereséshez, végsősoron hatékonyabb szoftverfejlesztést is eredményezhet.

4. Összefoglalás, konklúzió

A közelmúltban a szemmozgás követésen alapuló kutatások egyre inkább megjelentek olyan kognitív folyamatok vizsgálata tekintetében is, mint a programozás. A tanulmány egy helytelenül működő algoritmus hibáinak feltárása és kijavítása során a tesztalanyok tekintetének útvonalának megfigyelése útján keletkezett és rögzített szemmozgás paramétereket elemezte. A vizsgálat során néhány, a tesztalanyokra jellemző paraméter alkalmazásával, két csoport került kialakításra, ahol

az első csoport (*Try&R*) a hibakeresés során inkább az apróbb módosításokat és a gyakoribb fordítások és futtatások technikáját alkalmazta, amíg a második csoport tagjai (*Think&R*) nagyobb hangsúlyt fektettek az értelmezésre, továbbá kevesebb alkalommal éltek az alkalmazás fordítása és futtatása lehetőségével. A statisztikai kiértékelés során a két csoport szemmozgás követésére jellemző paraméterek (fixációk száma és átlagos fixációs időtartam) kerültek elemzésre. A kutatás eredményeként kijelenthető, hogy a kevesebb kapkodás, az alaposabb megfontoltság és figyelem esetében kevesebb információmennyiség feldolgozása is elegendő a hatékonyabb hibakereséshez és javításhoz, végsősoron hatékonyabb szoftverfejlesztést is eredményezhet. A szemmozgás paraméterek vizsgálata a jövőben támogató rendszerként szolgálhat a szoftverfejlesztők hatékonyságának elemzésében, a kognitív képességek fejlesztése, valamint az oktatásban a logikus, gondolkodás, átgondolt feladatvégrehajtásban egyaránt.

Köszönetnyilvánítás

A szerző ezúton szeretné megköszönni az Emberi Erőforrások Minisztériuma (EMMI) anyagi támogatását az NTP-NFTÖ-22-B-0003.

Irodalomjegyzék

- [1] Ujbanyi, T., Stankov, G., Nagy, B. : A transparent working environment in MaxWhere virtual space, In 2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), 2019, pp. 475-478, DOI: [CogInfoCom47531.2019.9089967](https://doi.org/10.1109/CogInfoCom47531.2019.9089967)
- [2] Demeter, R., Kővári, A. : Digitális szimuláció jelentősége a jövő társadalmát meghatározó mérnökök kompetenciafejlesztésében, Civil Szemle, 2020, vol. 17, no. 2, pp. 89–101
- [3] Molnár, G., & Benedek, A. : Three Dimensional Applications in Teaching and Learning Processes, Procedia-Social and Behavioral Sciences, 2015, vol. 191, 2667-2673, DOI: [10.1016/j.sbspro.2015.04.600](https://doi.org/10.1016/j.sbspro.2015.04.600)
- [4] Kovari, A. : Digital Transformation of Higher Education in Hungary in Relation to the OECD Report, In DIVAI 2022, 14th International Scientific Conference on Distance Learning in Applied Informatics, 2022, pp. 229–236
- [5] Benedek, A., Molnár, G. : Supporting the M-Learning Based Knowledge Transfer in University Education and Corporate Sector, International Association for the Development of the Information Society, 2014, pp. 339–343
- [6] Gógh, E., Racsó, R., Kővári, A. : Experience of Self-Efficacy Learning among Vocational Secondary School Students, Acta Polytechnica Hungarica, 2021 vol. 18, no. 1, pp. 101–119, DOI: [10.12700/aph.18.1.2021.1.7](https://doi.org/10.12700/aph.18.1.2021.1.7)
- [7] Guzsvinecz, T., Orbán-Mihálykó, É., Perge, E., Sik-Lányi, C. : Analyzing the spatial skills of university students with a Virtual Reality application using a desktop display and the Gear VR., Acta Polytechnica Hungarica, 2020, vol. 17, no. 2, pp. 35-56., DOI: [10.12700/aph.17.2.2020.2.3](https://doi.org/10.12700/aph.17.2.2020.2.3)
- [8] Guzsvinecz, T., Orbán-Mihálykó, É., Sik-Lányi, C., Perge, E. : Investigation of spatial ability test completion times in virtual reality using a desktop display and the Gear VR., Virtual Reality, 2022, vol. 26, no. 2, pp. 601-614., DOI: [10.1007/s10055-021-00509-2](https://doi.org/10.1007/s10055-021-00509-2)
- [9] Kővári, A. : Study of Algorithmic Problem-Solving and Executive Function, Acta Polytechnica Hungarica, 2020, vol. 17, no. 9, pp. 241–256., DOI: [10.12700/aph.17.9.2020.9.13](https://doi.org/10.12700/aph.17.9.2020.9.13)
- [10] Farkas, I., Dukan, P., Katona, J., & Kovari, A. : Wireless sensor network protocol developed for microcontroller based Wireless Sensor units, and data processing with visualization by LabVIEW, In 2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMi), 2014, pp. 95-98, DOI: [sami.2014.6822383](https://doi.org/10.1109/sami.2014.6822383)
- [11] Sik, D., Csorba, K., Eklér, P. : Toward Cognitive Data Analysis with Big Data Environment, In 9th IEEE International Conference on Cognitive Infocommunications, 2018, pp. 23-28, DOI: [coginfocom.2018.8639948](https://doi.org/10.1109/coginfocom.2018.8639948)
- [12] Benedek, A., Molnár, G., Szűts, Z. : Practices of crowdsourcing in relation to big data analysis and education methods, In 2015 IEEE 13th International Symposium on Intelligent Systems and Informatics (SISY), 2015, pp. 167-172., DOI: [10.1109/sisy.2015.7325373](https://doi.org/10.1109/sisy.2015.7325373)
- [13] Cepeda Porras, G., Guéhéneuc, Y. G. : An empirical study on the efficiency of different design pattern representations in UML class diagrams, Empirical Software Engineering, 2010, vol. 15, no. 5, pp. 493-522., DOI: [10.1007/s10664-009-9125-9](https://doi.org/10.1007/s10664-009-9125-9)
- [14] De Smet, B., Lempereur, L., Sharafi, Z., Guéhéneuc, Y. G., Antoniol, G., & Habra, N. : Taupe: Visualizing and analyzing eye-tracking data. Science of computer programming, 2014, vol. 79, pp. 260-278., DOI: [10.1016/j.scico.2012.01.004](https://doi.org/10.1016/j.scico.2012.01.004)
- [15] Jeanmart, S., Gueheneuc, Y. G., Sahraoui, H., Habra, N. : Impact of the visitor pattern on program comprehension and maintenance, In 2009 3rd International Symposium on Empirical Software Engineering and Measurement, 2009, pp. 69-78, DOI: [10.1109/esem.2009.5316015](https://doi.org/10.1109/esem.2009.5316015)
- [16] Sharif, B., Maletic, J. I. : An eye tracking study on the effects of layout in understanding the role of design patterns, In 2010 IEEE International Conference on Software Maintenance, 2010, pp. 1-10, DOI: [icsm.2010.5609582](https://doi.org/10.1109/icsm.2010.5609582)
- [17] Soh, Z., Sharafi, Z., Van den Plas, B., Porras, G. C., Guéhéneuc, Y. G., Antoniol, G. : Professional status and expertise for UML class diagram comprehension: An empirical study, In 2012 20th IEEE International Conference on Program Comprehension (ICPC), 2012, pp. 163-172., DOI: [10.1109/icpc.2012.6240484](https://doi.org/10.1109/icpc.2012.6240484)

- [18] Yusuf, S., Kagdi, H., Maletic, J. I. : Assessing the comprehension of UML class diagrams via eye tracking, In 15th IEEE International Conference on Program Comprehension (ICPC'07), 2007, pp. 113-122, DOI: [10.1109/icpc.2007.10](https://doi.org/10.1109/icpc.2007.10)
- [19] Busjahn, T., Schulte, C., & Busjahn, A. : Analysis of code reading to gain more insight in program comprehension, In Proceedings of the 11th Koli Calling International Conference on Computing Education Research, 2011, pp. 1-9, DOI: [10.1145/2094131.2094133](https://doi.org/10.1145/2094131.2094133)
- [20] Busjahn, T., Bednarik, R., Schulte, C. : What influences dwell time during source code reading? Analysis of element type and frequency as factors, In Proceedings of the Symposium on Eye Tracking Research and Applications, 2014 pp. 335-338, DOI: [10.1145/2578153.2578211](https://doi.org/10.1145/2578153.2578211)
- [21] Crosby, M. E., Stelovsky, J. : How do we read algorithms? A case study, Computer, 1990, vol. 23, no. 1, pp. 25-35., DOI: [10.1109/2.48797](https://doi.org/10.1109/2.48797)
- [22] Crosby, M. E., Scholtz, J., & Wiedenbeck, S. : The Roles Beacons Play in Comprehension for Novice and Expert Programmers, In PPIG, 2002, p. 5.
- [23] Seha, S., Papangelakis, G., Hatzinakos, D., Zandi, A. S., Comeau, F. J. : Improving eye movement biometrics using remote registration of eye blinking patterns, In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019, pp. 2562-2566, DOI: [10.1109/icassp.2019.8683757](https://doi.org/10.1109/icassp.2019.8683757)
- [24] Zhu, H., Salcudean, S. E., Rohling, R. N. : A novel gaze-supported multimodal human-computer interaction for ultrasound machines, International journal of computer assisted radiology and surgery, 2019, vol. 14, no.7, pp. 1107-1115., DOI: [10.1007/s11548-019-01964-8](https://doi.org/10.1007/s11548-019-01964-8)
- [25] Meng, J., Streitz, T., Gulachek, N., Suma, D., He, B.: Three-dimensional brain-computer interface control through simultaneous overt spatial attentional and motor imagery tasks, IEEE Transactions on Biomedical Engineering, 2018, vol. 65, no. 11, pp. 2417-2427., DOI: [10.1109/tbme.2018.2872855](https://doi.org/10.1109/tbme.2018.2872855)
- [26] Voßkühler, A., Nordmeier, V., Kuchinke, L., Jacobs, A. M. : OGAMA (Open Gaze and Mouse Analyzer): open-source software designed to analyze eye and mouse movements in slideshow study designs, Behavior research methods, 2008, vol. 40, no. 4, pp. 1150-1162., DOI: [10.3758/brm.40.4.1150](https://doi.org/10.3758/brm.40.4.1150)
- [27] Kovari, A., Katona, J., & Costescu, C. : Evaluation of eye-movement metrics in a software debugging task using gp3 eye tracker, Acta Polytechnica Hungarica, 2020, vol. 17, pp. 57-76., DOI: [10.12700/aph.17.2.2020.2.4](https://doi.org/10.12700/aph.17.2.2020.2.4)