

心拍変動の計測によるビジュアル型言語とテキスト型言語の 学習時の精神的ストレス分析について

梅澤 克之^{*1} 越川 拓海^{*2} 中澤 真^{*3} 平澤 茂一^{*4}

^{*1} 湘南工科大学 ^{*2} 旭情報サービス (株) ^{*3} 会津大学短期大学部 ^{*4} 早稲田大学

About the Mental Stress Analysis during Learning of Visual- and Text-based Programming Language by Measuring Heart Rate Variability

Katsuyuki Umezawa^{*1} Takumi Koshikawa^{*2} Makoto Nakazawa^{*3}
Shigeichi Hirasawa^{*4}

^{*1} Shonan Institute of Technology ^{*2} Asahi Intelligence Service Co., Ltd.
^{*3} Junior College of Aizu ^{*4} Waseda University

Our research project aims to establish a methodology for the transition between the two types of languages. In our previous research, we proposed an intermediate content with intermediate features between visual-based language and text-based language. And we showed that the comprehension of the text-based programming language is improved by inserting the intermediate content learning between the learning of both types of languages. In this study, we measure the heart rate variability of the learner during the learning of the visual-based programming language and the text-based programming language, and confirm whether there is any difference in the mental stress during the learning of both types of programming languages. Experiments confirmed that the heart rate variability value decreased (stress level increased) during visual-based language learning, especially in subjects accustomed to keyboard input.

キーワード：ビジュアル型言語，テキスト型言語，学習分析，心拍変動，精神的ストレス

1 はじめに

近年，プログラミングの入門としてビジュアル型のプログラミング言語（以降，ビジュアル型言語と呼ぶ）が使われるようになってきている．その後はC言語やJava言語などのテキスト型プログラミング

言語（以降，テキスト型言語と呼ぶ）に移行していくことになる．しかしシームレスな移行方法は確立されていない．

我々は，ビジュアル型言語からテキスト型言語への移行の方法論を確立することを目的とする研究プロジェクトを開始している．具体的には，ビジュア

ル型言語とテキスト型言語の学習の利点を有し、両者の差異を埋める教育コンテンツ（中間型言語とよぶ）を検討・試作し、実証実験を通して、学習結果だけでなく学習中の学習状態を評価する。これにより、その中間型言語の有効性を評価し、今後の初等・中等プログラミング教育に役立つ教育コンテンツ（中間型言語）を完成させる。ビジュアル型言語とテキスト型言語の溝を埋める想定の中間型言語の評価に関しては、今までは、学習後のアンケートや成績など、学習効果の結果のみに着目しており、理解できるようになったか否かを評価するのみであったが、これらの評価方法では、今回提案する中間型言語の効果を正確に測ることが出来ない。我々の研究プロジェクトでは、学習後の従来の評価方法に加えて、学習中の脳波や視線、表情などの生体情報および学習中の学習履歴を用いて学習状態を計測し、中間型言語がビジュアル型言語とテキスト型言語の中間的な役割を果たし、スムーズな移行に貢献しているかを分析・評価する。本研究が確立されると、プログラミング言語の初学者がビジュアル型言語による学習から始めて、その後、シームレスかつ自発的にテキスト型言語の学習に遷移できるようになることが期待できる。

我々は以前の研究⁽¹⁾で、上述の研究プロジェクトの一部として、まず中間コンテンツの提案に着目し、ビジュアル型言語の学習とテキスト型言語の学習の間に、我々が提案する中間コンテンツを用いた学習を挟む方がその後のテキスト型言語の理解度が高まるということを実証実験を通して示した。さらに、提案する中間コンテンツが、ビジュアル型言語とテキスト型言語の中間に位置づけられる特徴を有することをアンケートにより評価した。また、従来研究⁽²⁾では、ビジュアル型言語とテキスト型言語の学習の際に、学習者の生体情報（脳波、心拍、表情）を計測し、両言語の学習の際の生体情報に何らかの差異があるか否かの分析を行った。脳波を目的

変数、心拍と10種類の表情を説明変数として、重回帰分析を行ったところ、説明変数の係数の正負の符号に差異が見られた。

本研究では、従来研究⁽²⁾に引き続き、精神的ストレスの評価に向いているとされる心拍変動に着目し、ビジュアル型言語とテキスト型言語の学習の際のストレスの違いを評価する。

2 従来研究

Tóthら⁽³⁾は、ビジュアル型言語とテキスト型言語間のギャップの存在を指摘した。この研究ではビジュアル型言語（MIT App Inventor 2）からJava Bridge Code Generatorを知識伝達のメディエーターとして使用して、テキスト型言語（Android Studio）に移行する方法を検証した。この論文ではJava Bridge Code Generatorが、ビジュアル型言語とテキスト型言語の間のギャップを埋めるのに役立ったと主張している。

Weintropら⁽⁴⁾はビジュアル型言語から入ったが学習者とテキスト型言語から入った学習者がテキスト型言語に移行した後の知識移転の変化を実験で評価し、両者に有意差が無い事を示している。移行ではなくテキスト型言語のスキルを修得後の比較を行った。

我々は、以前の研究⁽¹⁾で、ビジュアル型言語の学習とテキスト型言語の学習の間に、我々が提案する中間コンテンツを用いた学習を挟む方がその後のテキスト型言語の理解度が高まるということを実証実験を通して示した。さらに、提案する中間コンテンツが、ビジュアル型言語とテキスト型言語の中間に位置づけられる特徴を有することをアンケートにより評価した。

また、我々は、脳波計を用いて学習者の学習状況を把握する方法の研究を推進中である。我々は、課題遂行中の脳波情報を取得し、難しい課題に取り組むと β/α の値が高くなることを示した⁽⁵⁾⁽⁶⁾。この研究の過程で、ビジュアル型言語（Scratch）の課題

に取り組んでいる際の脳波とテキスト型言語 (C 言語) に取り組んでいるときの脳波に差異があることを確認した。具体的には、ビジュアル型言語に関しては課題の難易度が高くなっても β/α の値が高くなることはなかった。これによりビジュアル型言語とテキスト型言語の学習過程には異なる思考が行われている可能性が示唆された⁽⁷⁾。

3 実験方法

3.1 実験参加者

今回の実験の参加者は、湘南工科大学の4年生の7名である。彼らは同じ大学の同じ学部でプログラミング関連のコースを数年間学んだ経験を有する。彼らのプログラミングスキルはほぼ同等である。ただし、分析の後でわかったことだが、キーボードの入力技術に関して学生間で差があった。

3.2 実験に使用する Web サービス

ビジュアル型言語のプログラミングを行う際は、Google Blockly⁽⁸⁾ (図1参照) を利用した。このサイトにはパズルや迷路などの課題が掲載されているが、今回はテキスト型言語と内容を合わせるために、音楽の課題に取り組むこととした。また、テキスト型言語のプログラミングを行う際は、JSFiddle⁽⁹⁾ (図2参照) を利用した。このサイトはテキスト型言語である JavaScript 言語を実行できる統合環境である。リソース設定として Beepplay ライブラリを追加することによって Beep 音によって音楽を作ることができる。

3.3 実験に使用する問題

まず、練習を1回行い、その後、本実験を2回行った。具体的には楽譜を見ながらその楽譜の通りの音ができるようにプログラミングを行う課題を行った。それぞれの練習と実験で対象とした歌は表1の通りである。なお、これらの楽譜を印刷し、実験時に実験参加者に提示した。なお、実験で取り扱った楽譜に

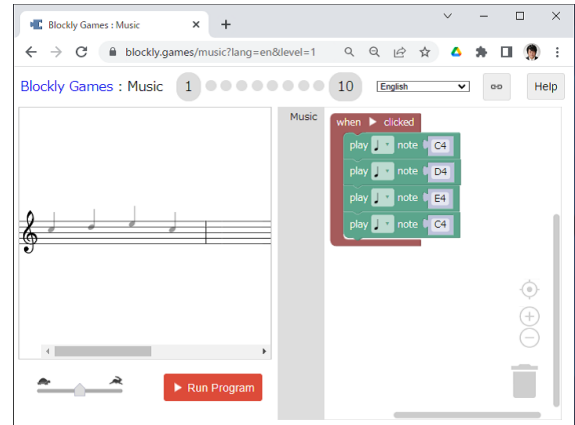


図 1: Google Blockly (Music) の画面

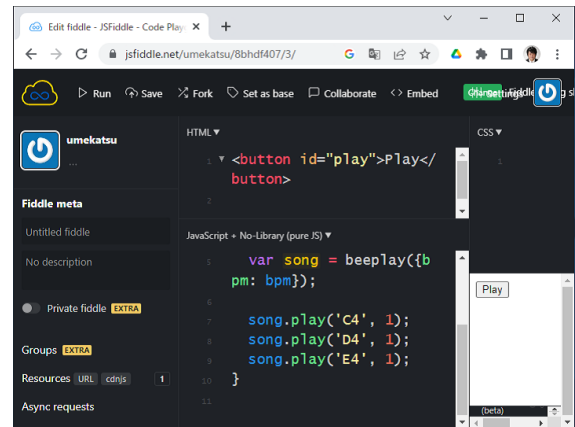


図 2: JSFiddle (Beeplay) の画面

は、音符の上に国際式表記 (C3, C4 など) が記入されている。図3~図5にそれぞれの歌の楽譜の一部を示す。

表 1: 対象とした歌

実験	曲名
練習	カエルの合唱
本実験 1 回目	メリーさんの羊
本実験 2 回目	ジングルベル

3.4 実験に使用する生体機器

3.4.1 心拍変動計

心拍変動の計測には Apple 社の “Apple Watch 8” を使用する。また、心拍変動の計測アプリとして W



図 3: かえるのうた (Froggy's Song) の楽譜



図 4: メリーさんのひつじ (Mary Had A Little Lamb) の楽譜

MS社 (WMS, Inc) の“心拍変動の分析・心電図アナライザー”を使用する。このアプリは Apple Watch と連携して HRV (root Mean Square of Successive Differences: rMSSD) を計算することができる。Apple Watch 8 で心拍変動を計測する場合、腕に装着した状態で別側の手の指を Digital Crown (リューズ部分) に当てる必要がある。30 秒間指を当てたままにすることで心拍変動の指標の一つである rMSSD が計測される。

3.4.2 キーロガー

マウスを多用するビジュアル型言語と、キーボードを多用するテキスト型言語で、それぞれの入力デバイスの使用頻度等を観測するために実験用 PC にキーロガーを組み込み、実験中の入力デバイスの使用状況を計測した。

3.5 実験の流れ

図 6 に実験の全体フローを示す。図 6 に示すように、テキスト型言語から取り組むグループ (グループ A) と、ビジュアル型言語から取り組むグループ (グループ B) に分けた。心拍変動の計測は、3.4.1 節に示したように、指を Digital Crown (リューズ部分) に添える必要があるため実験の最中 (プログラミング中) に計測することができない。そのため



図 5: ジングルベル (Jingle Bells) の楽譜

に図 6 に示すように実験の前後 30 秒間で心拍変動を計測することとした。なお、キーロガーによる入力情報はそれぞれの実験中に取得している。

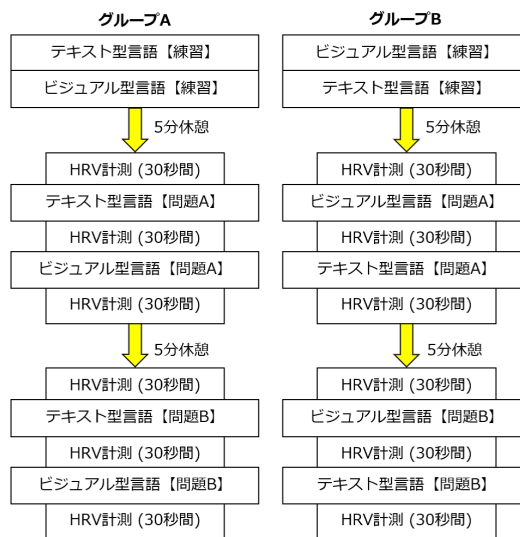


図 6: 実験の流れ

4 実験結果

グループ A (参加者 A~D) の心拍変動を図 7 に、グループ B (参加者 E~G) の心拍変動を図 8 に示す。実験の順番が心拍変動の変化に影響を与える懸念があったために、今回の実験ではグループ A とグループ B に分けて実験の順番を入れ替えた。図 7 と図 8 を見ると、実験参加者 C と G 以外は、どちらもグループの参加者もテキスト型言語の実験の前後で、心拍変動の値が上がっている。また、ビジュアル型言語に関しては、実験参加者 B 以外はどちらのグループの参加者も実験の前後で下がっている。一般的にストレスレベルが上がると心拍変動は下がり、

反対にストレスのバランスが保たれていれば心拍変動は上がると言われている。つまり、テキスト型言語では、多くの実験参加者はストレスバランスを保っているが、ビジュアル型言語では、多くの実験参加者はストレスレベルが上がっていると言えそうである。次節以降でこの結果が統計的に正しいと言えるか否かを検証する。

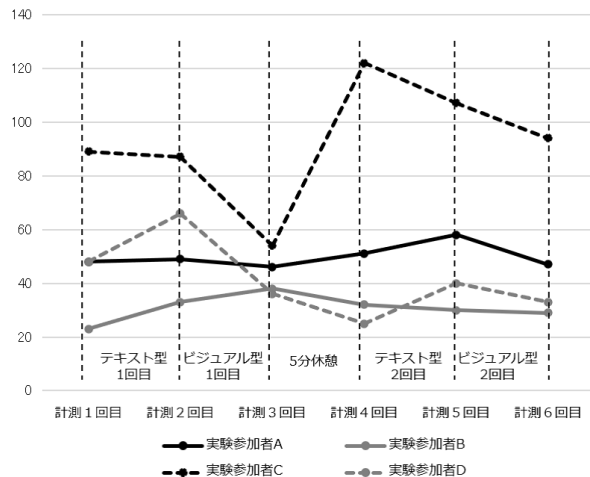


図 7: グループ A の心拍変動

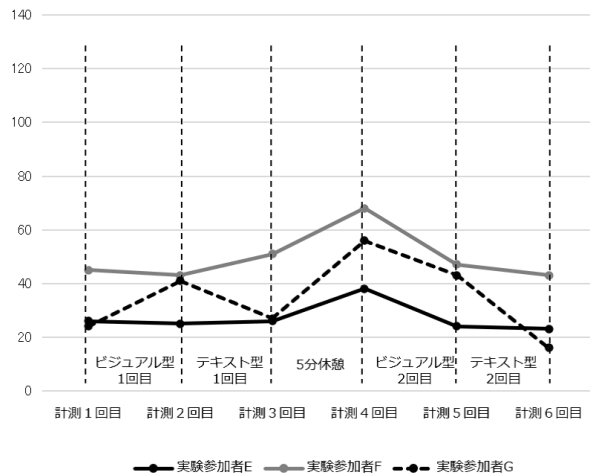


図 8: グループ B の心拍変動

5 分析

前節の図 7 と図 8 で示した通り、実験の順番が異なってもビジュアル型言語とテキスト型言語の実験の前後の心拍変動の変化の傾向に差異がなさそうである。よって本節以降の分析では、グループ A のテキスト型言語 (1 回目) の実験の前後の計測 (つまり計測 1 回目と計測 2 回目) と、グループ B のテキスト型言語 (1 回目) の実験の前後の計測 (つまり計測 2 回目と計測 3 回目) を統一的に扱う。他の計測に関しても、グループ A とグループ B の同種の実験の前後の計測を同じものとして扱う。

また、文献⁽¹⁰⁾によると、rMSSD の分布は正規分布には従わないが、rMSSD の値を自然対数変換をすることで正規分布に従うとされている。また文献⁽¹¹⁾によると、観測値の独立性、母集団分布の正規性、等分散性の前提があれば、サンプルサイズ (ただし $n \geq 2$) に関係なく t 検定などのパラメトリック検定が保証されるということである。さらに対応のある t 検定では等分散性の仮定は不要である。よって以降の分析では rMSSD を自然対数変換を行ったデータを使って対応のある t 検定を行っていく。

さらに文献⁽¹⁰⁾では、今回実験に用いたような 30 秒という短時間の計測による rMSSD などの HRV 指標の計測では、グループ間の統計量の検定を行う前に、同一グループ内で計測時間を延ばした実験結果と比較しなければならぬとされているが、今回の分析では、30 秒以上の計測を行っていないため同一グループ内の検定は省略する。

5.1 母平均の差に関する t 検定

全参加者を対象に、テキスト型言語 (1 回目)、ビジュアル型言語 (1 回目)、テキスト型言語 (2 回目)、ビジュアル型言語 (2 回目) のそれぞれの実験の前後で心拍変動 (rMSSD) の自然対数変換値の平均値に差が出ているかを検定する。対応のある t 検定を行った。検定結果を表 2 と図 9 に示す。

表 2: 心拍変動の平均値の検定

実験	前/後	平均値	p 値
テキスト型 (1回目)	前	3.7229	0.2604
	後	3.7899	
ビジュアル型 (1回目)	前	3.7552	0.3005
	後	3.6761	
テキスト型 (2回目)	前	3.7443	0.2811
	後	3.6418	
ビジュアル型 (2回目)	前	3.9580	0.0023**(< 0.01)
	後	3.7209	

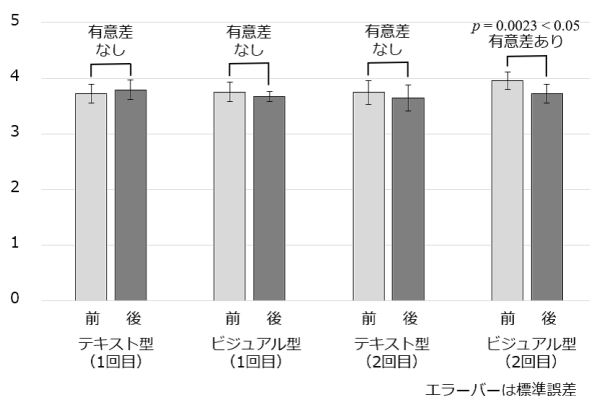


図 9: 心拍変動の平均値の検定

表 2 および図 9 に示した通り、ビジュアル型言語の 2 回目の実験の前後で心拍変動の値が有意に低くなった。その他に関しては統計的に有意な差とはならなかった。この結果より、テキスト型言語の学習はストレスバランスには影響がないが、ビジュアル型言語の学習では、ストレスレベルが上がったと考えられる。

5.2 キーボード入力経験の違いを考慮した心拍変動の平均値の分析

前節では、実験参加者全員を対象に平均値の差の検定を行った。本節では、キーボード入力経験の違いに着目する。今回の実験では、各課題の実行中にキーロガーによりキーボードとマウスの入力を監視した。そのログを見ると、テキスト型言語の課題の際に、実験参加者 A, E, F, G の 4 名は Ctrl-C (コ

ピー) と Ctrl-V (貼り付け) を多用していることがわかった。逆に実験参加者 B, C, D の 3 名は、それらのショートカットキーを全く使用していないことがわかった。このキーボード入力の経験が、今回のストレスレベルの変化に影響を与えているのではないかと考えた。Ctrl-C (コピー) と Ctrl-V (貼り付け) の多用者の検定結果を表 3 と図 10 に、Ctrl-C (コピー) と Ctrl-V (貼り付け) の不使用者の検定結果を表 4 と図 11 に示す。

表 3: Ctrl-C 多用者の心拍変動の平均値の検定

実験	前/後	平均値	p 値
テキスト型 (1回目)	前	3.6412	0.3695
	後	3.5944	
ビジュアル型 (1回目)	前	3.5337	0.2775
	後	3.6306	
テキスト型 (2回目)	前	3.6803	0.1983
	後	3.4324	
ビジュアル型 (2回目)	前	3.9857	0.0049**(< 0.01)
	後	3.6599	

表 4: Ctrl-C 不使用者の心拍変動の平均値の検定

実験	前/後	平均値	p 値
テキスト型 (1回目)	前	3.8318	0.1066
	後	4.0507	
ビジュアル型 (1回目)	前	4.0507	0.1532
	後	3.7367	
テキスト型 (2回目)	前	3.8295	0.3391
	後	3.9210	
ビジュアル型 (2回目)	前	3.9210	0.0618
	後	3.8024	

本結果より、テキスト型言語の課題解決時に Ctrl-C (コピー) と Ctrl-V (貼り付け) を多用しているためキーボードによる入力経験が高いと考えられる参加者は、ビジュアル型言語の課題解決時に心拍変動が統計的に有意に下がっていることが分かった。つまりマウス入力しかできないビジュアル型言語でストレスレベルが高くなってしまったということがわ

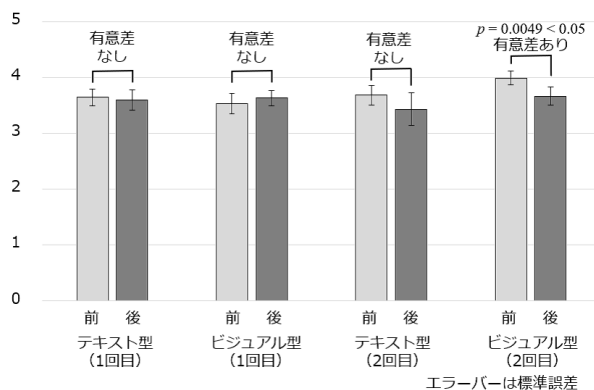


図 10: Ctrl-C 多用者の心拍変動の平均値の検定

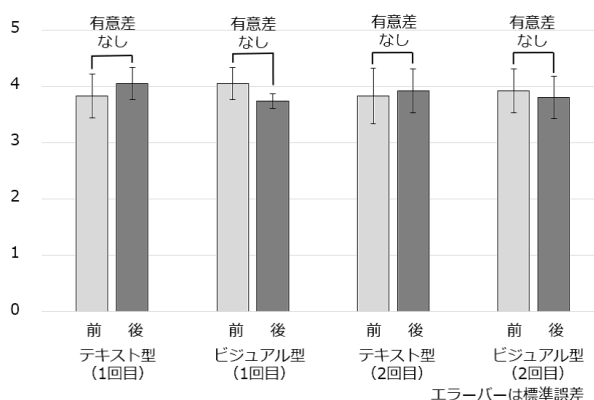


図 11: Ctrl-C 不使用者の心拍変動の平均値の検定

かった。

それに対して、テキスト型言語の課題解決時に Ctrl-C (コピー) と Ctrl-V (貼り付け) を全く使用していない参加者は、ビジュアル型言語の課題解決の前後で統計的に有意な差は見られなかった。

6 まとめと今後の課題

本研究では、ビジュアル型言語とテキスト型言語の学習の際に、両言語の学習の際の精神的ストレスに何らかの差異があるか否かを確認するために、学習者の心拍変動を計測した。ビジュアル型言語の学習時に心拍変動値が低くなる、つまりストレスレベルが高くなることを確認した。また、キーボード入力のログを取得することにより、キーボード入力に

慣れている被験者とそうでない被験者を別々に分析することによって、キーボード入力に慣れている被験者はビジュアル型言語の学習時によりストレスを感じていることを確認した。

今後、ビジュアル型言語とテキスト型言語間のギャップを埋める中間型言語を開発するにあたっては、今回の結論を踏まえて、様々な理解度や習熟度の学生に対してストレスを与えないような中間型言語を開発する必要があると考える。

研究倫理について

今回の実験は湘南工科大学研究倫理委員会の承認を得ている。また実験参加者から実験参加に関する署名を得ている。

謝辞

本研究は JSPS 科研費 JP22H01055, JP21K18535, JP20K03082 の助成を受けたものです。また、本研究の一部は、経営情報学会「ICT と教育」研究部会の助成を受けたものです。本研究成果の一部は早稲田大学理工総研プロジェクト研究「次世代 e-learning に関する研究」の一環として行われたものです。

参考文献

- (1) 梅澤克之, 石田昂大, 中澤真, 平澤茂一. ビジュアル型言語からテキスト型言語への移行のための中間コンテンツの提案と評価. 電子情報通信学会 (IEICE) 技術報告, pp. 1-7, 2022.
- (2) 梅澤克之, 中澤真, 平澤茂一. ビジュアル型言語とテキスト型言語の学習時の生体情報の比較について. 電子情報通信学会 (IEICE) 技術報告, pp. 1-6, 2023.
- (3) Tomáš Tóth and Gabriela Lovászová. Mediation of knowledge transfer in the transition from visual to textual programming. *Informatics in Education*, Vol. 20, pp. 489-511, 2021.
- (4) David Weintrop and Uri Wilensky. Transitioning from introductory block-based and text-based

- environments to professional programming languages in high school computer science classrooms. *Computers & Education*, Vol. 142, , 2019.
- (5) Katsuyuki Umezawa, Tomohiko Saito, Takashi Ishida, Makoto Nakazawa, and Shigeichi Hirasawa. Learning state estimation method by browsing history and brain waves during programming language learning. *Trends and Advances in Information Systems and Technologies. WorldCIST'18 2018. Advances in Intelligent Systems and Computing, Springer, Cham.*, Vol. 746, , 2018.
- (6) Katsuyuki Umezawa, Tomohiko Saito, Takashi Ishida, Makoto Nakazawa, and Shigeichi Hirasawa. Learning-state-estimation method using browsing history and electroencephalogram in e-learning of programming language and its evaluation. *Proceeding of the International Workshop on Higher Education Learning Methodologies and Technologies Online (HELMeTO 2020)*, pp. 22–25, 2020.
- (7) Katsuyuki Umezawa, Yutaka Ishii, Makoto Nakazawa, Michiko Nakano, Manabu Kobayashi, and Shigeichi Hirasawa. Comparison experiment of learning state between visual programming language and text programming language. *2021 IEEE International Conference on Engineering, Technology & Education (TALE)*, pp. 01–05, 2021.
- (8) Blockly games. <https://blockly.games/>. Accessed: 17 January 2023.
- (9) Jsfiddle. <https://jsfiddle.net/>. Accessed: 17 January 2023.
- (10) Leandro Pecchia, Rossana Castaldo, Luis Montesinos, and Paolo Melillo. Are ultra-short heart rate variability features good surrogates of short-term ones? state-of-the-art review and recommendations. *Healthcare Technology Letters*, Vol. 5, pp. 94–100, 2018.
- (11) 水本篤. サンプルサイズが小さい場合の統計的検定の比較 —コーパス言語学・外国語教育学への適用—. 統計数理研究所共同研究レポート 238, pp. 1–14, 2010.