

博士学位論文

携帯情報端末により取得したセンシングデータに基づいた
運動技能の学習支援に関する研究

東北大学大学院情報科学研究科
人間社会情報科学専攻
板垣翔大

博士学位論文目次

博士学位論文目次	3
博士学位論文図表目次.....	7
第1章 序論	11
1.1. 研究の背景.....	11
1.1.1. 運動技能とその学習	11
1.1.2. 情報通信技術を活用した運動技能の学習支援	13
1.1.3. 運動技能のセンシングの活用と課題.....	14
1.2. 問題の所在.....	15
1.3. 研究の目的.....	16
1.4. 本論文の各章の構成.....	17
第2章 先行研究.....	19
2.1. 運動技能の学習	19
2.2. 運動技能の分類	21
2.3. ICT を活用した学習支援.....	24
2.3.1. 対面による学習における学習支援	24
2.3.2. e-Learning による学習における学習支援	25
2.4. 運動技能のセンシング	27
2.4.1. 高機能な機器を用いたセンシング	27
2.4.2. センシング可能な携帯情報端末の普及	29
2.5. 対象とする運動技能であるかんな掛け	30
2.5.1. かんな掛けについて	30
2.5.2. 中学校技術・家庭科技術分野におけるかんな掛けの位置づけ	31
2.5.3. かんな掛けの動作の重要な観点.....	32
2.5.4. かんな掛けを対象とすることの妥当性.....	34

第3章	運動技能のセンシングデータを基にフィードバックを提示する	
	スマートフォンで動作する学習者用アプリケーションの開発と評価.....	35
3.1.	目的.....	35
3.2.	学習者用アプリケーションの開発.....	35
3.2.1.	学習者用アプリケーションの設計.....	35
3.2.2.	学習者用アプリケーションの概要.....	36
3.3.	専用かんなの開発.....	44
3.4.	学習者用アプリケーションの評価.....	51
3.4.1.	方法.....	51
3.4.2.	結果.....	53
3.4.3.	考察.....	57
3.5.	まとめと課題.....	57
第4章	運動技能のセンシングデータを基に指導参考情報を提示する	
	教師用システムの開発と評価.....	59
4.1.	目的.....	59
4.2.	教師用システムの概要.....	60
4.3.	教師用システムの評価.....	66
4.3.1.	方法.....	66
4.3.2.	結果.....	71
4.3.3.	考察.....	75
4.4.	まとめと課題.....	76
第5章	学習者用アプリケーションの模擬的な学習環境における評価.....	78
5.1.	目的.....	78
5.2.	方法.....	78
5.3.	結果.....	81
5.4.	考察.....	84
5.5.	まとめと課題.....	85

第 6 章	運動技能の学習を汎用の LMS で管理するためのプラグインの開発.....	87
6.1.	目的.....	87
6.2.	プラグインの開発.....	88
6.2.1.	プラグインの設計.....	88
6.2.2.	プラグインの概要.....	89
6.3.	プラグインを用いた学習コースの例	103
6.4.	他の運動技能への応用可能性	109
6.5.	まとめと課題.....	111
第 7 章	成果と今後の課題.....	113
7.1.	本研究の成果.....	113
7.2.	今後の課題.....	116
謝辞	118
出典	121
本論文に関する研究発表.....		129
付録	132
付録 1 :	授業実践 (第 4 章) で用いた質問紙.....	132
付録 2 :	開発したシステム等の主要なソースコード	133
2-1	学習者用アプリケーション (教師用システムとの連携時, かんな端末のメインのプログラム)	133
2-2	学習者用アプリケーション (Moodle との連携時, かんな端末のメインのプログラム)	146
2-3	学習者用アプリケーション (腰端末のメインのプログラム)	158
2-4	教師用システム (グラフ表示のプログラム)	160
2-5	教師用システム (リスト表示のプログラム)	163

2-7	教師用システム（学習者用アプリケーションからのデータを受け取る プログラム）	166
2-8	Moodle プラグイン（データベースに作成するテーブルの情報を 記すファイル）	168
2-9	Moodle プラグイン（問題の作成や編集を行うプログラム）	168
2-10	Moodle プラグイン（問題の作成時や編集時にデータベースへ アクセスするプログラム）	171
2-11	Moodle プラグイン（問題を表示するプログラム）	176
2-12	Moodle プラグイン（学習者用アプリケーションからのデータを受け取る プログラム）	177

博士学位論文図表目次

【第1章】

図 1-1 本論文の 各章の位置づけとその関係	18
-------------------------------	----

【第2章】

表 2-1 運動技能の分類.....	21
図 2-1 光学式モーションキャプチャシステムの構成図（八村 2007 より引用）	29
図 2-2 かんな	31
図 2-3 まっすぐに引くべきであることを示す図 （職業能力開発総合大学校能力開発研究センター 2009 より引用）	33

【第3章】

表 3-1 Nexus 5 のスペック	37
表 3-2 Galaxy Nexus のスペック	37
図 3-1 学習者用アプリケーションを使用している様子.....	38
表 3-3 評価の観点と試用しているセンサ.....	39
図 3-2 言葉による動作の結果とアドバイス.....	39
図 3-3 引きの速度および加速度のグラフ.....	40
図 3-4 引きの速度を再現するアニメーション	41
表 3-4 「③手元と腰の動き出しのタイミング」の観点における動作.....	42
図 3-5 学習者用アプリケーションの画面のフロー	43
図 3-6 開発した専用かんな	44
図 3-7 加工後のかんな台の寸法	45
図 3-8 専用かんなの端末固定部分.....	46
図 3-9 端末固定部 (A) の図面	47
図 3-10 端末固定部 (B) の図面	48
図 3-11 カバーの図面（厚さは 2 mm）	49
図 3-12 蝶番の図面	50
表 3-5 かんなの重量の比較	50

表 3-6	専用かんなの製作に必要な部品の一覧	51
図 3-13	学習者が練習している様子.....	52
図 3-14	「①引きの速度」の結果	54
表 3-7	「①引きの速度」の結果	54
図 3-15	「②引きの直進性」の結果.....	55
表 3-8	「②引きの直進性」の結果.....	55
図 3-16	「③手元と腰の動き出しのタイミング」の結果.....	56
表 3-9	「③手元と腰の動き出しのタイミング」の結果.....	56

【第4章】

図 4-1	システム全体の構成	60
図 4-3	学習者情報を登録する画面.....	61
図 4-4	グラフ表示	62
図 4-5	リスト表示	63
図 4-6	上達履歴表示	64
図 4-7	学習者用端末部に表示される上達履歴表示	65
図 4-8	教師用システムと連携時の学習者用アプリケーションの画面のフロー	66
図 4-9	学習者用端末部を用いて練習する生徒	67
図 4-10	指導参考情報提示部を用いて机間指導する教師.....	68
表 4-1	授業の流れ	69
表 4-2	質問紙の設問	70
図 4-11	「①引きの速度」の結果	71
表 4-3	「①引きの速度」の結果	71
図 4-12	「②引きの直進性」の結果.....	72
表 4-4	「②引きの直進性」の結果.....	72
図 4-13	「③手元と腰の動き出しのタイミング」の結果.....	73
表 4-5	「③手元と腰の動き出しのタイミング」の結果.....	73
図 4-14	質問紙調査の各設問の平均値.....	74

【第5章】

図 5-1	スマートフォンを取り付けたペットボトル	80
図 5-2	「①引きの速度」における得点の変化	81
表 5-1	「①引きの速度」における得点の変化	81
図 5-3	「②引きの直進性」における得点の変化.....	82
表 5-2	「②引きの直進性」における得点の変化.....	82
図 5-4	「③手元と腰の動き出しのタイミング」の結果.....	83
表 5-3	「③手元と腰の動き出しのタイミング」の結果.....	83

【第6章】

図 6-1	開発したプラグインの機能の概要.....	90
図 6-2	開発したプラグインの使用時の流れ.....	90
図 6-3	問題作成時の流れ	91
図 6-4	問題作成時に問題タイプを選択するページ	92
図 6-5	問題作成時に諸項目を入力するページ	93
図 6-6	各観点の問題で構成された小テスト	94
図 6-7	学習者による小テスト受験時の流れ.....	95
図 6-8	本プラグインを使用した小テストのページ.....	95
図 6-9	学習者用アプリケーションが算出した得点が入力された状態.....	96
コード 6-1	学習者用アプリケーションを起動するリンク	96
コード 6-2	学習者用アプリケーションでの「scheme」の指定	96
コード 6-3	学習者用アプリケーション起動時の Moodle からのデータの受け取り ..	97
コード 6-4	学習者用アプリケーションが算出した得点の Moodle への受け渡し	98
コード 6-5	学習者用アプリケーションが算出した得点の Moodle での受け取り	98
表 6-1	「numerical」からの修正.....	99
コード 6-6	「motorskill/db/install.xml」に加筆したソースコード	100
コード 6-7	「motorskill/edit_motorskill.php」に加筆したソースコード	101
コード 6-8	「motorskill/questiontype.php」に加筆したソースコード	101
コード 6-9	「motorskill/renderer.php」に加筆したソースコード.....	102
コード 6-10	「motorskill/posttomoodle.php」のソースコード	103

表 6-2	学習コースの一例	104
図 6-10	開発したプラグインを用いた学習コースの例	104
図 6-11	「2. かんなの握り方」を学習するページ	105
図 6-12	「6. 動作のポイント 3 : 手元と腰の動き出しのタイミング」を 学習するページ	106
図 6-13	「手元と腰の動き出しのタイミング」におけるフィードバック	107
図 6-14	学習コース例における学習のフロー	108
図 6-15	のこぎり引きを対象とした学習者用アプリケーション	109
コード 6-11	のこぎり引き動作を対象とした学習者用アプリケーション 起動時の Moodle からのデータの受け取り	110
コード 6-12	のこぎり引き動作を対象とした学習者用アプリケーションが 算出した得点の Moodle への受け渡し	110
図 6-16	のこぎり引きの 5 観点に対応した小テスト	111

第1章 序論

1.1. 研究の背景

1.1.1. 運動技能とその学習

人間が有する能力の1つに、運動技能が挙げられる。運動技能は、身体が目的にかなった動きをするように身体諸機能を調節する随意能力と定義されている（大築 1988）。自動車の運転やキーボード入力のタイピング、スポーツの技能、ものづくりにおける加工の技能など、日常生活においても多くの例を挙げることができる。運動技能は「人間のもつ能力として欠かすことができないもの」である（山内・春木 2001）。

それ故に、運動技能を習得するための学習も重要なものであるといえる。運動技能の学習について、Knapp（1963）は、指導を受けられず放任された初心者が、最初に思いついた方法を行い続ける傾向にあり、それが「轍」になると、その後指導を受けたとしても修正することが困難であることを指摘している。また、良い指導を与えられただけでは遂行できず、「やってみる」ことや試行錯誤することを繰り返す必要があると指摘している。その際、指導が与えられることにより、誤った動作を指摘したり、正しい動作をそのまま残したりすることで運動技能の学習が支援されるだろうと述べている。これらを整理すると、運動技能の学習方法は以下の2点にまとめられる。

1. 正しい運動技能について指導を受ける
2. 出来不出来に応じて指導を受けながら試行錯誤を繰り返す

こうした学習方法は身近に存在する。例えば自動車の運転の技能においては、基本的には一対一で、運転の技能について、教官が行う口頭や演示による指導を受ける。そして、学習者が自身の運転を助手席から教官に観察してもらい、指導を受けるのが一般的である。

また、学校教育における体育の授業や、中学校技術・家庭科技術分野の授業では、基本的には一対多で、教師が行う口頭や演示による一斉指導を受け、また、机間指導による観察や個別の指導を受けるのが一般的である。

その他の例としては、ものづくりなどにおける、いわゆる「職人」の世界での徒弟制度によるものが挙げられる。徒弟制度による学習では、学習者である弟子が一定期間親方の家に住み込み、雑用をしながら商工業のノウハウを見習い（武雄 2013）、「親方、師匠から弟子へ、個人から個人へ、職人技とそれを支える精神、生活態度の伝承が図られ」る（山

田 2009). これは、見て「盗む」、「体で覚える」というものであり、「親方や師匠、または先輩から『会話やことば』で懇切丁寧に教えられるというものではないが、教わりたい弟子が師のそばで見て学び、間違ったときに叱られるというものである(山田 2009). それぞれ、指導者と学習者の人数や、受けることができる指導の違いはあるが、状況や目的に応じて運動技能の学習が行われてきた。

運動技能の学習には、何を行うかを理解して言語的に戦略を考える「認知段階」、どのように行うか様々な戦略が試行錯誤される「連合段階」、手続きが自動化され、注意が減少し言語が不要になる「自動化段階」があることが知られている(嘉戸・伊藤 2008). また、運動技能の学習では、適切なフィードバックが与えられながら学習を繰り返すことが重要であるとされている(山内・春木 2001).

上で2点に整理した運動技能の学習方法は、「1. 正しい運動技能について指導を受ける」が認知段階に当たり、「2. 出来不出来に応じて指導を受けながら試行錯誤を繰り返す」が連合段階ないしは、連合段階から自動化段階への遷移に当たるといえる。また、このときに受ける指導が「フィードバック」に該当するといえる。このように捉えれば、上で2点に整理した運動技能の学習方法は妥当なものであるといえる。

しかし、この学習方法には困難点も挙げられる。武雄(2013)によれば、「技能は、カンやコツが必要で、五感を駆使し、経験を積むことで培われ、文字や数値での表現が難しい」ものである。そのため、「1. 正しい運動技能について指導を受ける」だけで、指導者に内在する暗黙知を学習者が適切に受け取り、理解することは容易ではないと考えられる。

さらに、授業のような一対多の学習環境においては特に、安彦ほか(1993)が「技能教科」について「40人でもいまだ多いと言えよう」と指摘しているように、40人ほどの学級では、学習者全員が十分なフィードバックを受けることが困難であると考えられる。また、指導者がその場に不在の環境では、解説や指導を受けることがそもそも不可能である。これらの困難点に対しては何らかの支援が必要である。

1.1.2. 情報通信技術を活用した運動技能の学習支援

運動技能の学習方法の困難点に対して、情報通信技術（Information and Communication Technology : ICT）を活用し、学習を支援する教材の開発や研究が行われている。ここでは主に、「対面による学習で行われる学習支援」と「e-Learningによる学習で行われる学習支援」に大別して述べる。

対面による学習で行われる学習支援について、例えば中学校技術・家庭科技術分野のデジタル教科書には、材料と加工に関する技術における、熟練者の模範的な木材加工の映像が含まれており、生徒に提示することができる（例えば、間田ほか 2011）。学習者が完全に理解するまで同じ動作を何度も繰り返して演示することは、教師にとって負担を伴う場合があるが、動画であれば何度でも繰り返し再生することができる。また、途中で一時停止させて観察することも可能となる。さらには、世界に有数の熟練者の技能であっても、動画に収録されていれば本人がその場に不在でも観察することが可能である。

しかし、通常の動画では、カメラが撮影したアングルからしか観察できず、必ず死角が生まれることが欠点であるといえる。この問題に対し、安藤・住川（2012）は、仮想空間を利用し、熟練者ののこぎり引き動作を観察することができる教材を開発した。仮想空間では学習者が自由に視点を移動させながら観察することができる。これにより、通常の動画教材よりも、学習者の理解が深まったことが報告されている。

一方、e-Learningによる学習で行われる学習支援について、例えば島（2007）は、大学の体育実技科目におけるフライングディスク競技の技能を対象とし、「テキストやイラスト、そして動画などを盛り込んだ電子教材」をCD-ROMで配布し、学生が各自で事前に学習してから授業に参加する実践を行った。その結果、「投げ方などを事前に知ることができる」、「あらかじめ見ておくと先生の説明が理解しやすかった」という意見が学生から得られた。

また、高ほか（2011）は、小児歯科学における治療の技能を対象に、実習書に準じたデモンストレーションに関する音声付きの動画やテキストを、実習開始の1か月前からWebで閲覧することができるようにした。その結果、「繰り返し見ることができる」、「事前にイメージできる」といった点で、実習における内容や技能に関する理解を深める上で有用であることが示唆された。

このように、対面による授業においても、e-Learningによる学習においても、ICTを活用した運動技能の学習支援が行われており、効果が認められている。しかし、これらは主に熟練者の運動技能やその説明を見たり読んだりして学習するものが多い。すなわち、認知段階の学習支援が中心であり、連合段階や自動化段階の学習支援までは十分に行われていないといえる。

石川ほか（2005）によれば、「熟達化には、自分自身を客観的に観察し、自分の知覚や行動の間違っている部分や無意識で行っていた部分などに『気づき』、『意識的に修正する』ことが重要」である。そこで、ICTを活用して学習者の運動技能をセンシングし、取得したデータを基に技能を上達させるためのフィードバックを提示することができれば、運動技能の学習における連合段階および自動化段階の学習も支援することができると考えられる。

1.1.3. 運動技能のセンシングの活用と課題

運動技能をセンシングする手法の代表的なものに、モーションキャプチャがある。佐藤ほか（2010）は、モーションキャプチャを用いて舞踊の技能をセンシングした。そして、取得したデータを基に作成したグラフとCGアニメーションを見せながらインタビューを行った結果、「講師はまっすぐ踊って、自分たちはまだまだ傾いていたことにはじめて気がついた」などの、新たな「気づき」や「理解」が得られたと報告されている。これは、運動技能の学習における連合段階および自動化段階を対象とした、ICTを活用した学習支援であるといえる。

運動技能をセンシングする手法は他にも存在する。安孫子ほか（2005）は、かんな掛けを対象とし、ハイスピードカメラで撮影した映像を分析し、熟練者と非熟練者の動作を比較し、非熟練者への指導法への示唆を得た。また、橋爪（2013）は、のこぎり引きを対象とし、動作時の筋電位を測定し、熟練者の技能の特徴を明らかにした。高井ほか（2016）は京壁塗りの技能を対象とし、動作の三次元測定、筋電位測定および眼球運動測定の結果から明らかとなった熟練者の技能の特徴に基づいたe-Learning教材を作成した。これらのように、運動技能をセンシングする手法は様々存在し、教育的利用への検討がなされてきた。

しかし、これらに必要な装置は、高価であることや大掛かりであることから、日常の学習には活用しづらいことが課題であるといえる。例えば、渡部（2007）が行った磁気式モーションキャプチャは、スタジオに磁界発生装置が2つ置かれ、被験者の関節間に11個の磁気センサが取り付けられた状態で行われた。光学式モーションキャプチャは、被験者の関節近くに20から30のマーカが取り付けられ、それを複数の高精度ビデオカメラで撮影する（八村 2006）。実際、佐藤ほか（2010）はモーションキャプチャを学習者の学習そのものに用いているが、2年間で4回の使用であった。安孫子ほか（2005）、橋爪（2013）、高井ほか（2016）では、センシングは日常の学習に用いられておらず、学習および指導上の参考となる知見を得るにとどまっている。日常の学習で用いることが可能で、連合段階および自動化段階の学習の支援を実現するためには、必要な装置は可能な限りコストをかけず、かつ簡便に使用できることが望ましい。

1.2. 問題の所在

前節で、従来行われてきた運動技能の学習方法では限界があることや、それに対してICTを活用した学習支援が行われていることを述べた。ICTを活用した学習支援については、運動技能のセンシングが認知段階の学習に役立てられていることや、一方でセンシングが日常の学習に用いられておらず、学習および指導上の参考となる知見を得るにとどまっております。これは、運動技能をセンシングするために必要な装置が高価で大掛かりであるために、日常の学習で用いられるほど身近でないことに起因していると考えられる。本研究ではこのことを問題として捉えた。可能な限りコストをかけず、かつ簡便に使用可能なセンシングの装置や手法の検討が必要であるといえる。

昨今では、スマートフォンやウェアラブルデバイスなどの携帯情報端末が身近に普及し、内蔵されたセンサを用いて心拍や体温、体動などをセンシングし、ヘルスケアに活用するサービスが実現されている（寺田 2015）。このような身近なデバイスを用いたセンシングは、運動技能の学習における連合段階および自動化段階の学習支援にも応用できるものと考えられる。

1.3. 研究の目的

本研究では、従来のセンシングの手法よりも低いコストで、かつ簡便に運動技能をセンシングし、取得したデータに基づいて、運動技能の学習における連合段階および自動化段階の学習を支援する方法を、システム等の開発および評価を通して検討することを目的とした。

運動技能のセンシングには、身近に普及しているデバイスであるスマートフォンを用いることとした。新機種購入に伴って自宅等に放置されている中古のスマートフォンも使用可能であると考えれば、十分に身近な機材であるといえる。

対象とする運動技能は、かんな掛けとした。かんな掛けは、熟練を要する技能であるが、従来「勘」や「コツ」に基づいた感覚的な表現による指導が行われてきた(田中ほか 1991)。そのため、センシングにより定量的に評価することに価値があると考え、対象とした。また、かんな掛けは全身を大きく使う運動技能であり、細かな運動技能に比べて、スマートフォンに内蔵されたセンサでも容易にセンシング可能であると考えられる。かんな掛けを対象とすることの妥当性等については、「2.5.対象とする運動技能であるかんな掛け」で詳述する。

運動技能の学習支援の方法の検討に当たっては、授業のような対面による学習とe-Learningによる学習の2つの視点から行うこととした。いずれの学習においても、学習者の運動技能をセンシングし、取得したデータを基にフィードバックを提示することには意義があるといえる。

対面による学習においては、その場に指導者が存在し、指導者からの指導を受けることができるが、1人の指導者が多数の学習者を指導する場合においては、指導者が個々の学習者に十分なフィードバックを与えることが困難であると考えられる。そのため、取得したデータを活用して指導者を支援することができれば、学習者の学習を間接的に支援することができると考えられる。

e-Learningによる学習においては、その場に指導者が不在であるため、センシングやフィードバックの提示は必須であるといえる。加えて、従来e-Learningによって学習されてきた認知段階の学習と、センシングやフィードバックの提示が取り入れられた連合段階および自動化段階の学習が組み合わせられることが望まれる。

1.4. 本論文の各章の構成

本論文は題目を「携帯情報端末により取得したセンシングデータに基づいた運動技能の学習支援に関する研究」とし、7章から成る。

「第1章 序論」では、研究の背景や問題の所在、および研究の目的を述べた。

「第2章 先行研究」では、運動技能の学習とその分類について、ICTを活用した学習支援について、運動技能のセンシングや身近に普及している携帯情報端末について、対象とする運動技能であるかな掛けについて、関連する先行研究を整理する。

「第3章 運動技能のセンシングデータを基にフィードバックを提示するスマートフォンで動作する学習者用アプリケーションの開発と評価」では、スマートフォンに内蔵されたセンサを活用して学習者の運動技能をセンシングし、取得したデータを基に技能を上達させるためのフィードバックを提示する学習者用アプリケーションの開発および評価について述べる。

「第4章 運動技能のセンシングデータを基に指導参考情報を提示する教師用システムの開発と評価」では、第3章で述べる学習者用アプリケーションを用いて練習した際のデータを収集し、指導参考情報として提示する教師用システムの開発について述べる。また、対面による学習として、中学校技術・家庭科技術分野における授業での実践を通して行った、学習者用アプリケーションおよび教師用システムの評価について述べる。

「第5章 学習者用アプリケーションの模擬的な学習環境における評価」では、第3章で述べる学習者用アプリケーションの模擬的な学習環境における評価について述べる。第3章および第4章での評価は、専用かなや削るための木材、削り台など必要なものが全て揃った「実践的な学習環境」におけるものであった。これに対し、「模擬的な学習環境」とは、必要なものを揃えることが困難である環境を指す。

「第6章 運動技能の学習を汎用のLMSで管理するためのプラグインの開発」では、第3章で述べる学習者用アプリケーションを用いた運動技能の学習を、汎用のLMSで管理するためのプラグインの開発について述べる。

「第7章 成果と今後の課題」では、本研究で得られた成果を整理し、今後の課題について述べる。

図 1-1は、本論文の各章の位置づけとその関係である。第3章と第4章では、授業のような「A1：対面による学習」の支援を検討する。第5章と第6章では、「A2：e-Learningによる学習」の支援を検討する。

第3章と第5章は、「B1：センシング・フィードバック」という点で対応している。第4章と第6章は、「B2：データの管理・活用」という点で対応している。

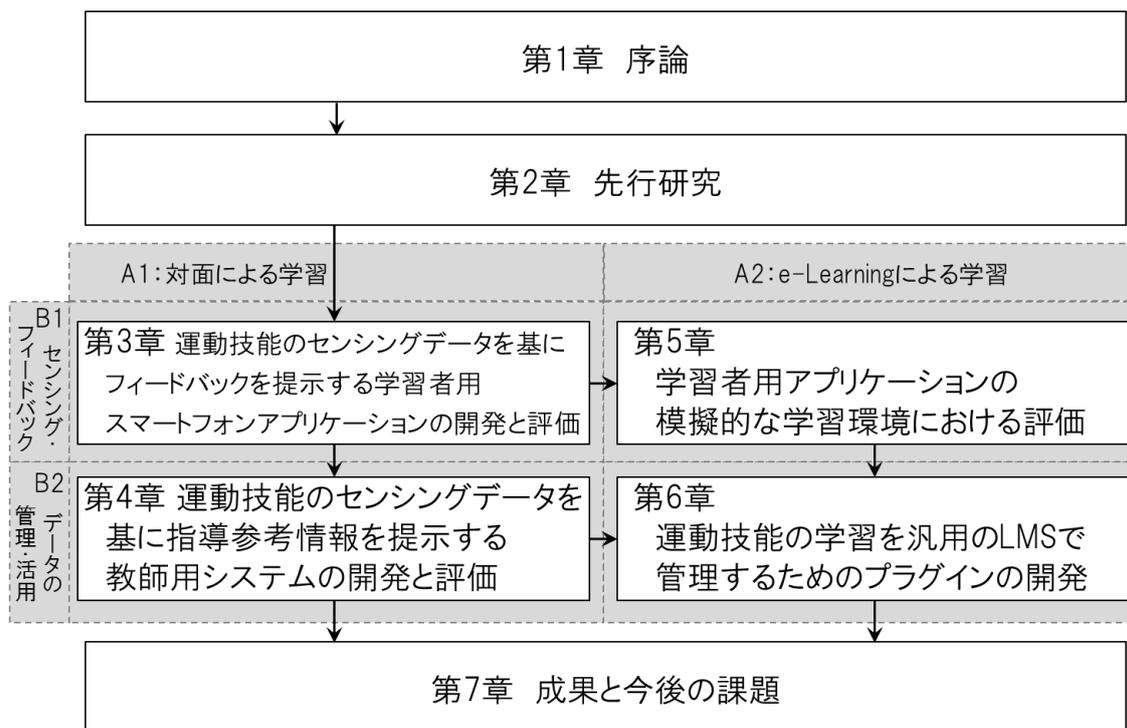


図 1-1 本論文の各章の位置づけとその関係

第2章 先行研究

2.1. 運動技能の学習

学習する運動技能について前もって解説を聞いておくこと（ガイダンス）が、運動技能の上達に対して重要である（山内・春木 2001）。松田・近藤（1969）は回転盤追跡課題において、「target は 1 秒 1 回の割合で回ってきます。それに合わせて円を描くように하십시오。」などの情報を音声で伝えた群の方が、同様の情報を伝えられなかった群よりも上達することを報告している。

また、Davies（1945）は、アーチェリーを対象に、矢の打ち方について指導を受けた群と受けなかった群の比較を行った。その結果、指導を受けた群の方が、指導を受けなかった群に比べて上達したことが確認された。また、指導を受けた群は与えられた指導を柔軟に取り入れたことに対し、指導を受けなかった群は、後に指導を与えられても、それを取り入れて試行錯誤することに消極的であったことが確認された。学習の初期段階においては特に、適切な指導を受けることが重要であるといえる。

運動技能の指導については、「注意を自分の身体運動そのものに向けさせる internal focus」と「運動が環境に及ぼす効果に向けさせる external focus」が存在する。また、「一般的には external focusの方がパフォーマンスの向上や学習にとっては有効である」とされている（谷 2012）。しかし、熟練度との関係でいうと、初心者には internal focusの方が、経験者には external focusの方が有効である（Natalie et. al. 2003）。かんな掛けでいえば、初心者には internal focusとして削りの動作に着目した指導が、経験者には external focusとして表面の仕上がりや削り屑の薄さに着目した指導が考えられる。

学習の過程で試行錯誤を繰り返す際には、フィードバックの提示が伴う必要がある。フィードバックは、内在的フィードバックと外在的フィードバックの2つに大別できる（谷 2012）。内在的フィードバックとは、「視覚、聴覚、固有感覚など、その情報源が学習者自身にある」とされており、学習者が自ら気づいて得る情報であるといえる。外在的フィードバックとは、「指導者や外部の機器など、その情報源が学習者の外部となる」とされており、学習者自身の気づきではなく外部から気づかされて得る情報であるといえる。本研究では「フィードバック」を、外在的フィードバックを指す言葉として用いる。

Bilodeau et. al. (1959) は、レバーを定められた大きさだけ動かして弧を正確に描く作業において、誤差の量をフィードバックすることの効果調べた。結果、フィードバックが与えられると誤差が小さくなり(学習の成立)、与えられないと学習が成立しないことが明らかになった。また、途中からフィードバックの提示が中断されると、その時点から学習が成立しなくなることも明らかになった。運動技能の学習において、フィードバックが与えられ続けることが重要であるといえる。

Trowbridge and Cason (1932) は、3 インチ長さの直線を引く作業において、フィードバックの有無やフィードバックの内容が学習に及ぼす効果を調べた。具体的には、(1) 誤差がフィードバックされた群、(2) 誤差が 1/8 インチ内だったら「正」、それ以上だったら「誤」とフィードバックされた群、(3) フィードバックが提示されなかった群、(4) 無意味なフィードバック(ランダムなアルファベット3文字)が提示された群の上達を比較した。(1)の群が最も上達した。次いで上達したのが(2)の群であった。無意味なフィードバックが提示された(4)の群は、何もフィードバックが提示されなかった(3)の群よりも成績が悪かった。このことから、学習は与えられるフィードバックの内容や質に影響されることが明らかになった。

Greenspoon and Foreman (1956) は、3 インチ長さの直線を引く作業において、運動技能を実行してからフィードバックが提示されるまでの時間の遅延が学習に及ぼす影響について調べた。その結果、遅延なしでフィードバックが提示された群が最も上達した。また、フィードバックの提示が遅延されるほど成績が悪かったことが確認された。フィードバックが提示されなかった群は上達が認められなかった。

これらの先行研究から、運動技能の学習においては、提示されるフィードバックが適切なものであり、かつ即時的であることが重要であるといえる。

石黒ほか(2013)は、フィードバックする情報を操作し、運動技能における偽薬(プラセボ)効果が与える影響を明らかにした。具体的には、複数回の握力の測定において、「最高握力値より値が小さい時に、本来の値よりも最高握力値に近い値に加工」して提示する「維持」と、最高握力値より小さい時に、本来の値よりも最高握力値から遠い値に加工して提示する「挑発」の2種類の情報操作を行った。「維持」では、パフォーマンスが低下していないと自信を持たせ、「挑発」では、パフォーマンスが低下したと思ひこませて奮

起させることが情報操作の意図であった。結果、「維持」と「挑発」の両方において、情報操作をしなかったときに比べ、握力が低下しないことが明らかになった。提示されるフィードバックがパフォーマンスに強い影響を与えていることや、フィードバックが提示されることの重要性がわかる。

運動技能の学習の過程は、前述の通り、言語的に戦略を考える「認知段階」、どのように行うか、様々な戦略が試され、試行錯誤する「連合段階」、手続きが自動化され、注意が減少し言語が不要になる「自動化段階」がある。この3つの段階において、ガイダンスは認知段階にあたるといえる。また、適切で即時的なフィードバックを提示されながらの試行は連合段階であり、それを繰り返すことにより、自動化段階へ移るといえる。ただし、これらの各段階は、連続的に向上するため、明確な境界はないとされている（嘉戸・伊藤 2008）。

2.2. 運動技能の分類

先行研究から挙げられる運動技能の分類を表 2-1 に整理した。ここでは、動作の大きさから分類したもの、実行する環境から分類したもの、運動技能の連続性から分類したものの3つについて述べる。

表 2-1 運動技能の分類

分類の視点	分類		出典
動作の大きさ	粗大運動	微細運動	(Knapp 1963)
	(大) ←————→ (小)		
実行する環境	閉鎖技能	開放技能	(シュミット 1994)
	(安定) ←————→ (不安定)		
連続性	分離的技能	系列技能	(シュミット 1994)
	(なし) ←————→ (あり)		

動作の大きさの視点から分類したものに「粗大運動」と「微細運動」がある (Knapp 1963)。「粗大運動」とは、主として大きな筋肉群を使うものであり。「微細運動」とは、運動の大きさが小さく、小さな筋肉に活動が集中するものである。例えば、「1.1.3. 運動技能のセンシングの活用と課題」で述べた舞踊の動作は、「粗大運動」に含まれ、「微細運動」には、キーボードのタイピングや箸の使用などが含まれる。ただし、粗大運動と微細運動は明確には区別できないことには注意が必要である。例えば、キーボードのタイピングをする際には、指先の細かな動きが当然必要であるが、椅子に座り、机に向かうという大きな筋肉群の制御も必要となるように、多くの運動技能において、粗大運動と微細運動の両方の要素を有するためである。

本研究では、センシングするためのデバイスとしてスマートフォンを用いるが、粗大運動であればスマートフォンほどの大きさのデバイスを身につけてセンシングすることが可能であるといえる。一方、微細運動の場合、スマートフォンを手指に装着し、細かな動きをセンシングすることは現実的ではなく、異なるセンシングの手法が検討されるべきであると考えられるため、研究の限界と判断し、本研究では対象外とする。

運動技能を実行する環境の視点から分類したものに「閉鎖技能」と「開放技能」がある (シュミット 1994)。「閉鎖技能」とは、比較的安定した、予測可能な環境で行われる運動技能である。「開放技能」とは、予測しにくい、変化する環境で行われる運動技能である。閉鎖技能の例としては、ゴルフのスイングや水泳などが挙げられる。開放技能の例としては、サッカーの試合やボクシングなど、相手やボール等の状態によって即座に対応しなければならないものが挙げられる。Knapp (1963) はこのことについて、必要な動作を実行するだけでなく、状況やニーズに応じて正しい意思決定を行うことも運動技能に含まれるとしている。

閉鎖技能は、実行する環境が安定している分、理想的な技能を定めやすく、上達のためのフィードバックの提示も行いやすいと考えられる。一方、開放技能は実行する環境によって望ましい技能が変化するため、閉鎖技能とは異なる評価やフィードバックの提示が必要と考えられる。故に、状況に応じた意思決定を含む開放技能までを包括して一般化することは困難であると考えられるため、本研究では閉鎖技能のみを対象とし、開放技能は対象外とする。

運動技能の連続性の視点から分類したものに「分離的技能」、「系列技能」および「連続的技能」がある（シュミット 1994）。「分離的技能」とは、技能の開始と終了が明確な技能である。「連続的技能」とは、逆に開始と終了が明確ではない、繰り返される技能である。この両者の間に「系列技能」がある。分離的技能の例としては、投球やゴルフのスイングなどが挙げられる。連続的技能の例としては、歩行や水泳などが挙げられる。

分離的技能は、運動の開始と終了が明確である分、評価をしやすいと考えられる。また、終了したタイミングでフィードバックを提示することが可能である。一方、系列技能や連続的技能では、連続しているために評価の開始と終了のタイミングや、フィードバックを提示するタイミングが分離的技能とは異なると考えられる。これらを全て包括するよう一般化することは困難であると考えられるため、研究の限界と判断し、系列技能と連続的技能は本研究では対象外とする。

以上を整理すると、本研究での対象は、「粗大運動」、「閉鎖技能」、「分離的技能」の全てに該当する運動技能である。いずれか1つでも該当しないものは、センシングや評価、フィードバックの提示等において異なる手法が必要と考えられるため、研究の限界とし、対象外とする。

本研究で対象とする運動技能であるかんな掛けは、かんなの握り方などの微細運動の要素も含まれるが、削りの動作としては、立ち上がった状態で全身を使って削るため、粗大運動の要素が強いといえる。また、固定された木材を削る安定した環境での技能であるため、閉鎖技能に分類される。さらに、削り始めと削り終わりが明確であるため、分離的技能に分類される。

また、運動技能は身体動作に加え、その結果生まれる成果が伴う場合がある。例えば、かんな掛けにおける削る動作と削られた木材や、アーチェリーにおける射る動作と放たれた矢などである。成果が重要であることは言うに及ばないが、本研究では、その成果を生むための身体動作も重要であると捉えている。望ましい成果が得られれば身体動作がどのような状態でも構わない、という考え方もできるが、特に初心者にとっては、正しい身体動作を学習することが重要であるといえる。このことは、「2.1.運動技能の学習」で挙げた Davies (1945) による、アーチェリーにおいて指導を受けた群の方が受けなかった群より高得点を獲得した（高い成果を上げた）という知見や、Natalie et. al. (2003) による、

初心者には *internal focus*、経験者には *external focus* の指導が効果的であるという知見からも主張できる。本来、成果まで含めた評価を行うことが望まれるが、研究の限界として成果の評価は対象外とし、本研究では身体動作に焦点化することとした。

2.3. ICT を活用した学習支援

2.3.1. 対面による学習における学習支援

対面による学習における ICT を活用した学習支援として、初期のものに、CAI (Computer Assisted Instruction) が挙げられる。CAI は「コンピュータがもつ通信と高速計算・データベース検索、また電子テキスト、GUI (Graphical User Interface)、マルチメディアなどの特有の機能特色を生かし、コンピュータが教育と学習を支援するシステム」である (安彦ほか 1993)。

我が国では 1970 年代から学校現場での CAI の実践に力が注がれ始めた (東原 2008)。茨城県桜村立竹園東小学校では、算数や理科を対象に、1977 年から CAI の実践研究が行われた (中山・東原 1986)。ここで開発されたシステムは、一斉学習下において、いかにして個人のペースで、個人の能力に見合った学習を実現するかを目指したものである。このシステムでは、「教師用ディスプレイ」が設けられ、授業者が学級全体の学習の進行状況を確認し、他に比べて進行が遅れている学習者を把握することができた。

しかし、CAI による学習は、コンピュータが設置された教室でしか行うことができず、環境依存性が高かった。その後、技術が発展し、持ち運び可能なデバイスが活用可能となり、石塚ほか (2007) が、児童が一人一台の PDA (Portable Digital Assistant) を使用できる環境で授業実践を行った。実践で用いたソフトウェアでは、児童が PDA を用いて学習したデータが、教師用 PC にリアルタイムで表示されるようになっていた。この実践からも「教師は指導が必要な児童を全体から見出し、個別指導を行うことができた」ことが報告されている。さらに、PDA を用いることにより、教室内の特定の場所だけでなく、教室外、学校外へ持ち出しての実践も行われた。ICT がさらに発展した 2010 年前後からは、タブレット PC などの携帯情報端末が用いられている (文部科学省 2014)。

これらの先行研究の成果から、対面による学習において、学習者は各自が使用する機器やその上で使用できる教材があることで学習を行うことができるといえる。また、教師にとっては、中山・東原 (1986) でいう「教師用ディスプレイ」のような、個々の学習者の学習状況を教師が確認できる仕組みがあることで、教師による指導が支援され、その結果、

学習者が効率的な指導を受けることができるといえる。そしてこれらの学習支援は、環境依存性の低い持ち運び可能な携帯情報端末が用いられることにより、さらに多様な方法での学習支援が期待できるといえる。

2.3.2. e-Learning による学習における学習支援

e-Learning は、e ラーニングコンソシアム (2008) によれば、狭義には「いわゆる WBT といわれるもので、インターネットまたはイントラネットを利用してコンテンツ (教材) の配信が行われる」とされ、広義には「衛星通信、テレビ会議、あるいは CD-ROM、さらには各種電子機器による学習も含まれる」とされている。高岡 (2012) によれば、「場所や時間を選ばずに自由に学習できる環境を指す。狭義にはネットワークを利用して学習する環境を、広義には ICT を活用した学習方法全般を指している」とされている。広義には、前項で述べた CAI 等による学習も含まれるであろう。本研究では狭義な意味で e-Learning を捉えている。

e-Learning は、アメリカでは 1990 年代に普及し (高岡 2012)、日本では 2000 年が「e-Learning 元年」といわれている (e ラーニングコンソシアム 2008)。いまや学習の方法や環境として一般的になっているといえる。e-Learning による学習が一般的になった背景には、PC 等の機器の普及や通信インフラの整備の他に、e-Learning の長所が学習の支援に役立っていることが挙げられる。高岡 (2012) は、e-Learning の長所として、いつでもどこでも学習できる、個人の進捗度に合わせて学習できる、必要な内容を必要なときだけ学習できる、理解度の把握が容易、などを挙げている。また、e-Learning の活用の場面として、学校における学習の支援や、企業における社員のスキルアップや内定者の入社前教育、地方行政による住民向けの生涯学習や安全教育、遠隔教育での活用などを挙げている。特に昨今では、e-Learning は、急速に変化する社会の中で、学校を卒業した社会人が 24 時間いつでも学び、キャリアアップを図るための環境としても一層の充実が求められており (教育再生実行会議 2015)、現代の学習方法として不可欠なものであるといえる。

e-Learning では、学習管理システム (Learning Management System : LMS) と呼ばれるプラットフォームが用いられることが多い。LMS は、「『コンテンツの Web 配信・管理』『掲示板システム』『受講者管理と学習の進捗・成績管理』『教材コースの登録・管理』を基本機能としてもっている」とされている (植野 2007)。この中で、「コンテンツの Web

配信・管理」としては、「教材としてテキストのみではなく、画像・音声・動画などを用いたマルチメディア・コンテンツを Web を通じて配信できる」。また、「演習問題を提示して学習者の知識を確認し、理解できていない部分を同定できるしくみをもっている。特に理解できていない部分が即時にフィードバックされることが教材の理解を深めるうえで有効である。(中略) 演習問題に誤った場合、正解と解説を提示するだけでなく、該当コンテンツへジャンプできる機能(リンク)などが有用である」(植野 2007)。「受講者管理と学習の進捗・成績管理」としては、「学習者が学習したコンテンツの系列や学習所要時間、演習問題の結果や所要時間などを詳細にデータベースに蓄積できる機能があり、これらのデータを学習者や教師にフィードバックし、学習の進捗などを管理できる」(植野 2007)。コンテンツ単体によっても学習は可能であるが、このような機能を利用できることは、LMS によって管理された e-Learning の利点であるといえる。

代表的な LMS の 1 つに Moodle が挙げられる。Moodle は 1999 年に開発が開始された LMS であり、オープンソースで配布されている。日本語を含む 100 か国語以上の言語に対応しており、インストールも比較的容易であるため、広く利用されており、日本の大学で利用されている LMS の中では最もシェアが高い(大澤・中西 2015)。Moodle では、機能拡張のために定められている API により、多様なプラグインによる機能の拡張が行われていることも特徴の 1 つであるといえる。(赤倉・柏原 2016)。2017 年 11 月現在、公式ページ (<https://moodle.org/plugins/>) に掲載されているプラグインは 1400 を超える。小テストで用いられる様々な出題形式の問題を作成するためのプラグインや、画面上のユーザインタフェースや表示機能を提供するプラグインなど、目的に応じた様々な種類のプラグインの開発が可能である。プラグインは、Moodle 内部の機能の拡張だけでなく(例えば、中西ほか 2015)、外部のシステムと連携し機能を拡張することも可能である(例えば、横田ほか 2011)。

e-Learning には課題も存在する。板垣・堀田(2016)は、日本教育工学会における e-Learning に関連する論文等を調査し、e-Learning での学習の対象を、マルザーノの分類体系(黒上・泰山 2013)に沿って、「情報」、「心的手続き」、「精神運動手続き」に分類した。その結果、「情報」や「心的手続き」を対象とした論文等に比べ、「精神運動手続き」を対象とした論文等が極端に少ないことを指摘した。わずかにみられた「精神運動手続き」を対象とした論文等は、楽器の演奏技能等を対象に、熟練者の映像を見て動作を学習する

ものや、キーボード入力を対象に、速さや正確さを評価し、向上させようとするものであった。前者は認知段階の学習に当たるものである。後者は連合段階および自動化段階の学習であるが、微細運動であり、また入力された成果から評価する external focus である。

「2.2.運動技能の分類」で述べた本研究で狙いとするような、粗大運動や internal focus に該当する先行研究は見られなかった。

堀田・木原（2008）は、「学力向上を目指す ICT 活用」を、ブルームの教育目標の分類学に基づいて、認知的領域、情意的領域、精神運動的領域の3領域で整理した。そのうち、精神運動的領域の「熟練的動作・非論証的伝達」については、「ICT 活用はなじまない」としており、板垣・堀田（2016）の調査結果とも一致する。次節で述べる運動技能のセンシングを活用し、取得したデータを基に技能を上達させるためのフィードバックを提示することができれば、運動技能の学習における連合段階および自動化段階の学習も支援することができると考えられる。

2.4. 運動技能のセンシング

2.4.1. 高機能な機器を用いたセンシング

様々な運動技能を対象に、熟練者の運動技能の特徴の解明や学習のために、その動作等をセンシングした研究は数多くみられる。

佐藤ほか（2010）は、日本舞踊を対象に、磁気式モーションキャプチャを活用し、動作の三次元測定を行った。舞踊の動作をモーションキャプチャによりセンシングし、その結果から生成したグラフやCGアニメーションを見ることにより、自身の未熟な点に気付くことが可能になることが示唆された。池元ほか（2016）は、自動車の修理における塗装作業時のスプレーガンの使用を対象に、光学式モーションキャプチャを活用し、動作の三次元測定を行った。熟練者と非熟練者の動作のそれぞれの測定結果に基づいた比較を通して、塗装対象と体との間の最適な距離や、その距離を実現するための理想的な姿勢を明らかにした。安孫子ほか（2005）は、かんな掛けを対象に、ハイスピードカメラを用いて動作の三次元動作測定を行った。熟練者と比較したときの非熟練者の動作の特徴を明らかにし、非熟練者への指導法への示唆を得た。これらの手法により動作をセンシングし、定量的に分析することの価値が見て取れる。

運動技能のセンシングの対象は動作だけではない。橋爪（2013）は、木材加工におけるのこぎり引きを対象に、筋電図測定装置を用い、肩関節および肘関節の筋活動を測定した。

熟練者と非熟練者の比較を通し、熟練者のみに見られる特徴を明らかにした。高井ほか(2016)は、京壁塗りを対象に、動作の三次元測定と合わせて、筋活動の測定や眼球運動の測定を行い、熟練者と非熟練者の比較を行った。そして、そこから得られた知見をe-Learning教材の開発に役立てた。澤井ほか(2013)は、漆塗りの基本動作の1つである平面の中塗り動作を対象に、眼球運動や脳波、心電図などを測定し、習熟過程における「間」や「按配」の変化を定量的に評価できることを示唆した。個人がもっている暗黙知を、動作の側面からだけでなく、筋活動や眼球運動、脳波などの、目視だけでは確認が困難な側面からも定量的に分析することは価値があるものであると同時に、ICTを活用することの利点であるといえる。

これらのような、運動技能の特徴を定量的に分析することができる様々なセンシングの手法は、運動技能の学習にも生かすことができると考えられる。しかし、上述した様々なセンシングの手法は、必要な装置が高価であることや大掛かりであることなどの欠点も挙げられる。佐藤ほか(2008)は、モーションキャプチャの欠点について以下のように述べている。

例えば、光学式モーションキャプチャならば、専用の服を着て、体にマーカーをつけなければならない。磁気式であれば、体にセンサーを取り付けなければならない。またモーションキャプチャは装置が大規模であり、どこにでも移動できるものではなく、基本的には専用のスタジオで撮影が行われるなどの制約が多い。また、モーションキャプチャは、専門家の領域であることは否めず、コンピュータ等に詳しくないものが簡単に使えるものではない。さらに、一般的に非常に高価である。

図 2-1は、八村(2007)が示す光学式モーションキャプチャシステムの構成図である。センシングするにあたり、複数のカメラやコンピュータ等が必要であり、佐藤ほか(2008)が指摘するように大規模であることが伺える。

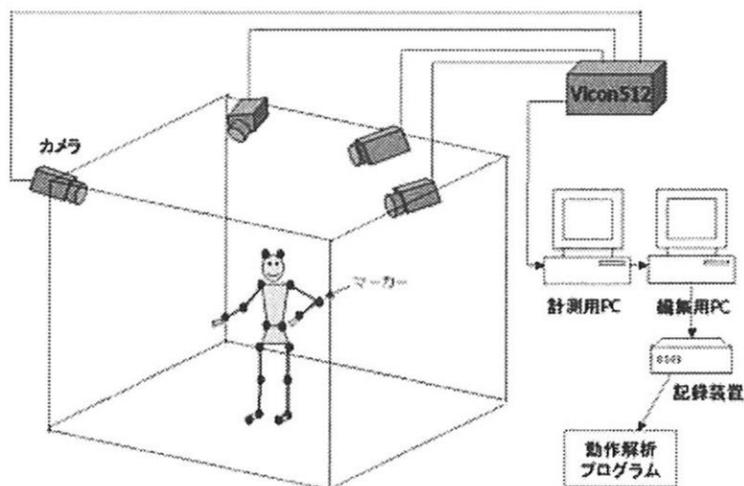


図 2-1 光学式モーションキャプチャシステムの構成図（八村 2007 より引用）

上述した先行研究の成果を踏まえれば、運動技能のセンシングの手法は、運動技能の学習支援に有効に活用できると考えられるが、このような欠点を鑑みると、特に日常の学習において、学習支援のために使用することは現実的であるとはいえない。従来のセンシングの手法よりも低いコストで、かつ簡便に使用できる手法の開発が望まれる。このような手法が開発されれば、運動技能の学習における連合段階および自動化段階の学習支援が実現できると考えられる。

2.4.2. センシング可能な携帯情報端末の普及

スマートフォンやウェアラブルデバイスなどの、各種センサを搭載した携帯情報端末が身近に普及している。総務省（2017）によれば、2016年時点での我が国におけるスマートフォンの世帯別の保有状況は71.8%である。2010年時点で9.7%であったことを踏まえると、急速に普及していることが確認できる。個人保有率では、20代で94.2%、30代で90.4%、13～19歳で81.4%に上り、若い世代では大半の人がスマートフォンを所有していることがわかる。

このように、広く普及しており多様な機能を備えたスマートフォンは、モバイルラーニング（Mobile Learning, m-Learning）と呼ばれているように既に学習のためのツールとしても用いられている（例えば、渡辺ほか 2014）。小学校、中学校および高等学校においては、スマートフォンの持ち込みや使用が制限されている現状があるが、村田・藤本（2013）が主張するように、学習者によって目的外の使用がされなければ十分活用可能であると考

えられる。

スマートフォンは、いわゆる従来の携帯電話としての機能以外にも多数の機能を備えている。例えば、端末の状態を検知するためのセンサが内蔵されている点が挙げられる。加速度センサやジャイロセンサなど、端末の動きをセンシングするものや、照度センサや近接センサなど、周囲の環境をセンシングするものなど、機種によって異なるが様々なセンサが内蔵されている。また、キャリアの回線以外に、Wi-Fi や Bluetooth などの通信機能を備えており、複数のスマートフォン同士での通信や外部機器との通信も可能である。そして、スマートフォンアプリケーションの開発によって、センサや通信機能などを利用可能であることもスマートフォンの特徴であるといえよう。これらを活用する専用のアプリケーションの開発により、運動技能のセンシングや、取得したデータに基づいたフィードバックの提示、取得したデータの送受信が可能になり、運動技能の学習支援に活用することができると考えられる。

運動技能の学習支援を目的としたものではないが、内蔵されたセンサを活用し、人の行動をセンシングするアプリケーションは既に存在する。例えば、「Google Fit」(<https://play.google.com/store/apps/details?id=com.google.android.apps.fitness>) は、万歩計の機能や、ランニングやサイクリングのスピードやルート等を記録する機能を備えている。「Sleep Cycle alarm clock」(<https://itunes.apple.com/jp/app/sleep-cycle-alarm-clock/id320606217>) は、スマートフォンに内蔵された加速度センサで睡眠時の体の動きを検出して、睡眠の深さを測定し、目覚めの良いタイミングでアラームを鳴らす。特にヘルスケア分野においては、スマートフォンやウェアラブルデバイスに内蔵されたセンサを用いて心拍や体温、体動などをセンシングしヘルスケアに活用するサービスが実現されている(寺田 2015)。このような身近なデバイスを用いたセンシングは、運動技能の学習における連合段階および自動化段階の学習支援にも応用できるものと考えられる。

2.5. 対象とする運動技能であるかな掛け

2.5.1. かな掛けについて

かな掛けとは、かな (図 2-2) を用いて木材を薄く削り、表面を仕上げる技能である。なお、様々な種類のかなや削り方が存在するが、中でも最も一般的なかなである「平かな」を用いた「平削り」の技能を本研究では対象とする。

かな掛けは、木材加工の中では、切る、削る、叩く、掘る、組み立てる、といった基

礎的な作業のうちの「削る」にあたる基本的な運動技能である（藤沖ほか 2013）。また、かんな掛けは製作の仕上がりに大きく影響する重要な運動技能である（安孫子 1997）。

一方で、かんな掛けは従来「勘」や「コツ」に基づいた感覚的な表現による指導が行われてきた熟練を要する難易度の高い運動技能でもある（田中ほか 1991）。西洋のかんなは金属製であり、かつ押しで使用するため、被削材の上にかんなを押し付けやすく、また等速移動させやすい。これに対して日本のかんなは、精緻な加工のために木製であり、かつ引いて使用する構造になっており、難易度は高い（大工道具研究会 2009）。

例えば、「2.1.運動技能の学習」で挙げたような先行研究では、回転盤追跡課題や特定の長さの直線を引く課題など、比較的容易な運動技能が対象とされたのに対し、かんな掛けという難易度の高い運動技能を対象とした学習支援を検討することに、かんな掛けを対象とする意義があるといえる。



図 2-2 かんな

2.5.2. 中学校技術・家庭科技術分野におけるかんな掛けの位置づけ

かんな掛けは、学校教育では中学校技術・家庭科技術分野で扱われる。中学校学習指導要領解説の技術・家庭編（文部科学省 2008b）の「第2節 技術分野」の「2 技術分野の内容」の「A 材料と加工に関する技術」の「(2) 材料と加工法について、次の事項を指導する」には、「イ 材料に適した加工法を知り、工具や機器を安全に使用できること」とあり、また「使用する工具や機器については、刃物の形状を観察しやすい工具を取り上げ、

切削や切断の仕組みに気付かせ、工具や機器に適した材料の固定方法や安全な操作方法を知ることができるようにする」、「正しい使用方法とともに、姿勢、目の位置、工具などの持ち方、力配分など、作業動作の要素も関連することに気付かせる」とある。中学校学習指導要領（文部科学省 2008a）やその解説の技術・家庭編（文部科学省 2008b）では、木材題材を扱うことや特定の工具を用いることは明記されていないが、全日本中学校技術・家庭科研究会（2013）によれば、95.1%の中学校において、「材料と加工に関する技術」の題材として木材が扱われている。また、現在販売されている技術・家庭科技術分野の教科書では、3社全てが、「材料と加工に関する技術」において、主として木材題材を取り上げており、かんな掛けについても記載している（安東ほか 2016, 佐竹ほか 2016, 田口ほか 2016）。

2.5.3. かんな掛けの動作の重要な観点

田中・安孫子（1990）は、身体重心の移動を熟練者、経験者、初心者の動作の比較を通して分析した。結果として、熟練者ほど重心の移動を使って、全身で引いていることが明らかとなった。また、引く速度に関しても、熟練者ほど速く引いている様子が確認された。引きの速度や、手元のみでない全身の動きに言及している先行研究は他にも挙げられる。安孫子ほか（2005）はデジタルカメラを用いた三次元動作分析で、熟練者と非熟練者の動作を比較した。比較の結果、熟練者には初速度が速く、引き終わりに意識的にかんなを制止させる動作や、腰の引きを使った動作が見られた。

橘田ほか（2004）は、中学生のかんな掛けの動作を筋電位と画像解析から分析した。結果として、かんな切削の初速を大きくして一気に引く、かんなを押さえつけることより引く力を有効に発揮させる、上肢に依存するより、下肢を踏ん張って体重移動を有効に利用する、削り終わり時はかんなを押さえ、落とさずに水平に移動するよう支配する、の4点を身に付けた生徒が効果的なかんな掛けを行うことが明らかになった。

また、橘田ほか（2003）は、ビデオカメラでの撮影と筋放電測定により、かんな台の握り位置が切削動作に及ぼす影響について分析し、台の中央部を握るよりも、台尻部分を握るよう指導すべきであるとの知見を得た。その際、分析の観点として、平均切削速度、切削面の精度（傷出現数、削り屑厚さ、削り残し面積）、上肢および下肢の筋電位分布の3つを用いた。

以上の研究で得られた知見は、熟練者と非熟練者の動作における違いは、引きの速度、

腰や体重移動など全身を使った動作，引き終わりのかんなの支配の3つにまとめられる。本研究では，このうち，全ての先行研究で述べられていた，引きの速度と腰や体重移動など全身を使った動作の2つが特に重要であると考えた。

また，現在市販されている技術・家庭科技術分野の教科書（安東ほか 2016，田口ほか 2016，佐竹ほか 2016）や，木材加工の技能に関する書籍（木材加工教育研究会 1983，職業能力開発総合大学校能力開発研究センター 2009）には「まっすぐに」や「丁寧に」といった記述や，まっすぐに引くべきであることを示す図（図 2-3）が共通して見られる。かんなを引くときの直進性の重要性がうかがえる。

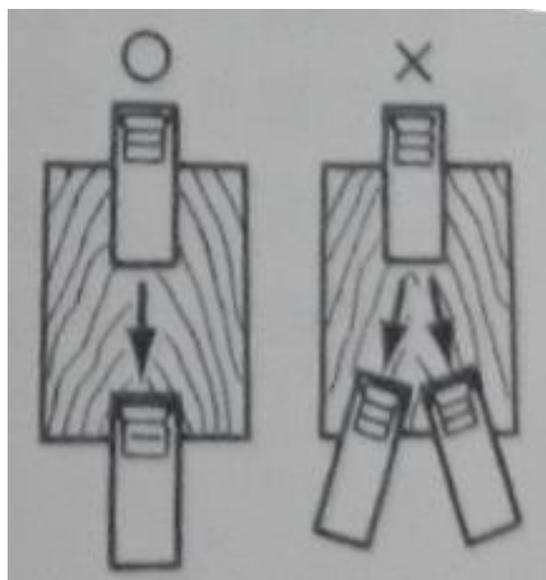


図 2-3 まっすぐに引くべきであることを示す図

（職業能力開発総合大学校能力開発研究センター 2009 より引用）

以上より，「①引きの速度」，「②引きの直進性」，「③手元と腰の動き出しのタイミング」の3観点が必要であると考えた。「①引きの速度」および「③手元と腰の動き出しのタイミング」に関する事項は，多くの先行研究で扱われている観点であり，熟練者と非熟練者の差が明らかなものであるため，評価・改善することでかんな掛けの上達が可能であると考えられる。「②引きの直進性」については，ただ力任せに引くだけでは，切削面が美しく仕上がらず傷や削り残しが出るため，速く引きつつもまっすぐに引く必要があるといえる。

2.5.4. かなな掛けを対象とすることの妥当性

対象とする運動技能の分類については「2.2.運動技能の分類」で述べた通りであるが、ここでは、その分類の中からはかなな掛けを対象とすることの妥当性を、Johnson (1961) による運動技能の4つの要素に基づいて述べる。

Johnson (1961) によれば、運動技能には「form」、「accuracy」、「speed」、「adaptability」の4つの要素が存在する。ただし、ここでいう「adaptability」とは、いかなる環境においてもその運動技能を実行できる「適応性」を意味している。かなな掛けを含む、比較的安定した予測可能な環境で行われる閉鎖技能においては、「adaptability」は存在しないといえよう。

かなな掛けでいう「①引きの速度」は「speed」に該当するといえる。「②引きの直進性」は、引くときの正確さや丁寧さという意味で、「accuracy」に該当するといえる。「③手元と腰の動き出しのタイミング」は、腰が動いてから手元が動く、という「型」であるため、「form」に該当するといえる。

以上のように、かなな掛けは閉鎖技能としての運動技能の要素をすべて有している。そのため、対象とする運動技能としての代表性は十分であると判断できる。また、「①引きの速度」、「②引きの直進性」、「③手元と腰の動き出しのタイミング」の3観点は運動技能の要素それぞれに対応しており、評価の観点も妥当であると判断できる。

第3章 運動技能のセンシングデータを基にフィードバックを提示する スマートフォンで動作する学習者用アプリケーションの開発と評価

本章では、かんな掛けを対象とし、従来のセンシングの手法よりも低いコストで、かつ簡便な手法として、スマートフォンを用いて学習者の運動技能をセンシングし、取得したデータを基にフィードバックを提示する学習者用アプリケーションの開発および運動技能の学習における連合段階および自動化段階の学習支援に対する評価について述べる。

3.1. 目的

前章までに、運動技能の学習支援として、ICTを活用して運動技能をセンシングして取得したデータを基にフィードバックを提示することを、従来のセンシングの手法よりも低いコストで、簡便に実現することの必要性を述べた。また、これが実現することにより、運動技能の学習における連合段階および自動化段階の学習支援を実現できることを述べた。

本章では、かんな掛けを対象とし、身近に普及しているスマートフォンを用いて、学習者の運動技能をセンシングし、取得したデータを基にフィードバックを提示する学習者用アプリケーション（以下、学習者用アプリケーション）を開発することとした。また、学習者用アプリケーションを用いて学習者のかんな掛けをセンシングするために、スマートフォンを装着することができる専用のかんな（以下、専用かんな）を開発することとした。そして、これらを用いた実験により、スマートフォンという身近なデバイスを用いて、運動技能の学習における連合段階および自動化段階の学習支援が可能か評価を行うこととした。

3.2. 学習者用アプリケーションの開発

3.2.1. 学習者用アプリケーションの設計

ICTを用いてかんな掛けの学習を支援した先行研究として、田中ほか（1991）の「かんな削り訓練装置」が挙げられる。これは、拡散反射式光電スイッチを用い、学習者のかんな掛けの速度をセンシングし、熟練者の速度と比較してフィードバックを提示する装置であった。この装置を用いた実験からは、「従来よりも短い練習時間で熟練者と同じような一定速度によるかんな削りができるようになった」という結果や、装置を使用したことで「ただ漫然と削るのではなく、評価されることで学習目標を常に維持できる」、「熟練者との速

度の違いを比較できるので、自分の欠点がどこか理解できる」といった感想が実験参加者から得られた。

これを参考に、本研究でも熟練者との比較や、即時的にフィードバックを提示する機能をもったスマートフォンアプリケーションを開発することとした。

かんな掛けの評価の観点とは、「2.5.2. かんな掛けの動作の重要な観点」で述べた以下の3観点とした。

- ① 引きの速度
- ② 引きの直進性
- ③ 手元と腰の動き出しのタイミング

「①引きの速度」および「②引きの直進性」は、手元の動きをセンシングすることにより評価できる。「③手元と腰の動き出しのタイミング」は、手元の動きに加えて腰の動きもセンシングする必要がある。そのため、2台のスマートフォンを用い、手元と腰の2か所をセンシングすることとした。

なお、「②引きの直進性」は学術的には的確な表現であるが、学習者にとってはわかりにくい表現であると考えられるため、学習者用アプリケーション内では「かんなのブレの少なさ」と表現することとした。

3.2.2. 学習者用アプリケーションの概要

学習者用アプリケーションは、Android OS 2.3.3以上のスマートフォンで動作する。スマートフォンの最低限の要件は、加速度センサおよびジャイロセンサを搭載していること、Wi-Fi および Bluetooth による通信が可能なことである。近年の機種の大半はこの要件を満たしている。

本研究で用いた機種は、LG エレクトロニクス社の Nexus 5、およびサムスン電子社の Galaxy Nexus であった。国内でのそれぞれの発売時期は、Nexus 5が2013年11月、Galaxy Nexus が2011年11月であった。それぞれの詳細なスペックを表 3-1 および表 3-2 に示した。

表 3-1 Nexus 5 のスペック

OS	Android4.4 → Android5.0 (アップデート)
CPU	Qualcomm Snapdragon™ 800, 2.26 GHz
RAM	2 GB
ストレージ	16 GB
サイズ	69.17 × 137.84 × 8.59 mm
重量	130 g
ディスプレイ	4.95 インチ フル HD (1920×1080) 静電容量式, マルチタッチ
通信	Wi-Fi (2.4G/5G) 802.11 a/b/g/n/ac Bluetooth 4.0 LE
センサ	加速度センサ, GPS, 電子コンパス, 近接/周囲光, ジャイロセンサ, 気圧計
バッテリー	2,300 mAh

表 3-2 Galaxy Nexus のスペック

OS	Android4.0 → Android4.2.2 (アップデート)
CPU	Texas Instruments OMAP 4460, 1.2 GHz
RAM	1 GB
ストレージ	16 GB
サイズ	68 × 136 × 8.9 mm
重量	138 g
ディスプレイ	4.7 インチ HD (1280×720) 静電容量式, マルチタッチ
通信	Wi-Fi (2.4G/5G) 802.11 a/b/g/n/ac Bluetooth v3.0
センサ	加速度センサ, GPS, コンパス, 近接センサ, ジャイロセンサ, 気圧計
バッテリー	1,750 mAh

図 3-1 は、学習者用アプリケーションを使用している様子である。2 台のスマートフォンが使用されている。1 台は「3.3.専用かんなの開発」で後述する専用かんなに装着されており（以下、かんな端末）、もう 1 台は腰付近のポケットに入れられている（以下、腰端末）。両端末は Bluetooth 接続によって連携する。両端末のペアリングはあらかじめ済ませておく。腰端末でセンシングされたデータは全てかんな端末へ送信される。かんな端末では、腰端末から受信したデータと、かんな端末でセンシングしたデータの全てを集計し、学習者へフィードバックを提示する。

学習者用アプリケーションでは、モデルとなる熟練者のかんな掛けのデータを登録することができる。学習者が練習を行うと、そのかんな掛けがセンシングされ、あらかじめ登録された熟練者の動作データと 3 観点において比較され、学習者にフィードバックが提示される。

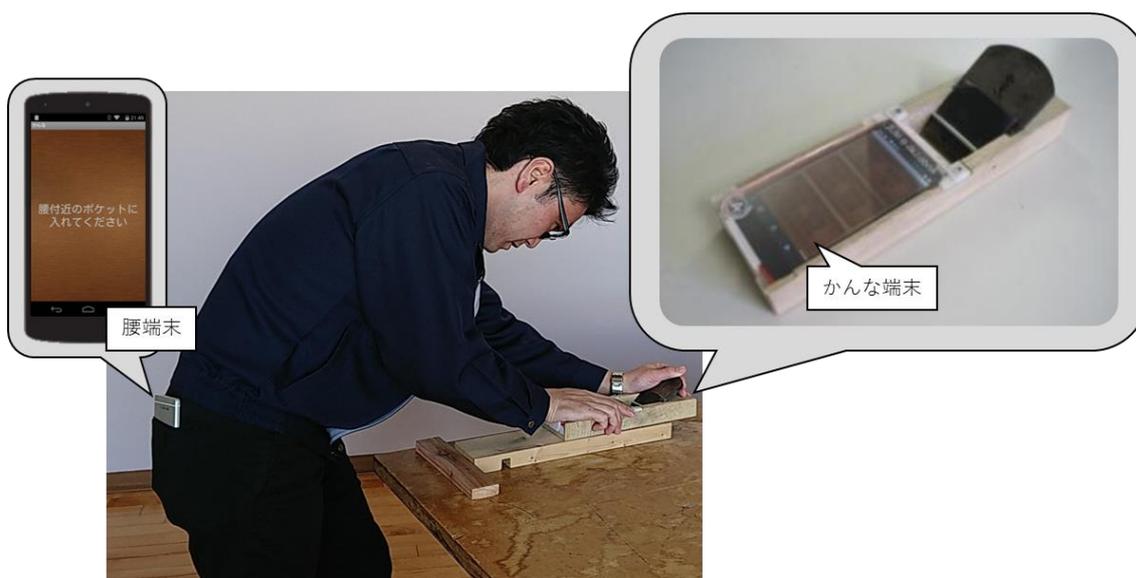


図 3-1 学習者用アプリケーションを使用している様子

表 3-3 は、3 観点のそれぞれをセンシングするために用いているセンサをまとめたものである。「①引きの速度」のセンシングには、かんな端末の加速度センサを用いている。「②引きの直進性」のセンシングには、かんな端末のジャイロセンサを用いている。「③手元と腰の動き出しのタイミング」については、両端末の加速度センサから得られる値の変移から、それぞれが動き出したタイミングを算出している。

表 3-3 評価の観点と試用しているセンサ

評価の観点	使用しているセンサ
① 引きの速度	かんな端末の加速度センサ
② 引きの直進性	かんな端末のジャイロセンサ
③ 手元と腰の動き出しのタイミング	両端末の加速度センサ

学習者用アプリケーションが提示するフィードバックには3つの表示がある。

1. 言葉による動作の結果とアドバイス
2. 引きの速度および加速度のグラフ
3. 引きの速度を再現するアニメーション

「1.言葉による動作の結果とアドバイス」(図 3-2)では、あらかじめ登録された熟練者の動作データの引きの速度や直進性を100点とした場合の学習者の得点や、その得点に応じたランクが「★」の数で表示される。またその結果を踏まえて、次回以降の練習で留意すべき事項が表示される。得点の具体的な算出方法は後述する。

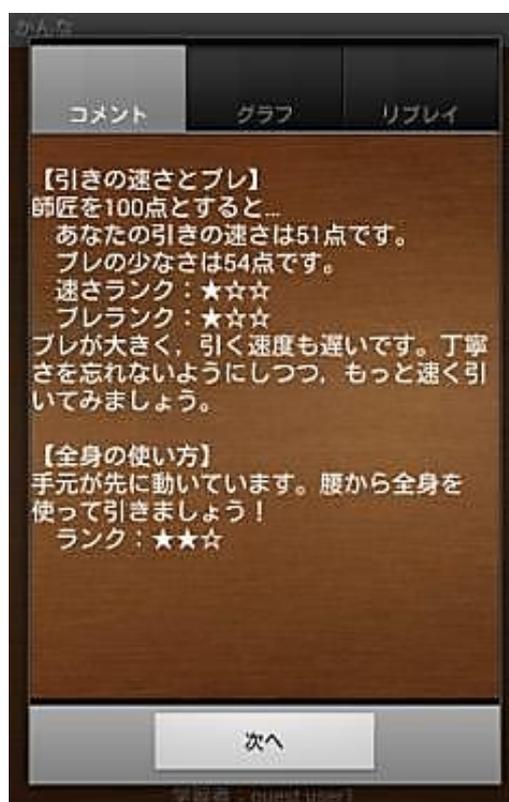


図 3-2 言葉による動作の結果とアドバイス

「2.引きの速度および加速度のグラフ」(図 3-3)では、練習を行った際の速度(図 3-3 上)や加速度(図 3-3 下)のグラフ(青色)が表示される。その際、あらかじめ登録された熟練者の動作データのグラフ(黄色)と重ね合わされて表示される。学習者は熟練者との違いを視覚的に確認することができる。また、腰が動き出したタイミングが横軸上に「○」で表示される。図 3-3 のようなグラフが表示された場合には、黄色で示された熟練者の動きは、幅が狭く速度が大きいことに対し、青で示された学習者の動きは、幅が広く速度が小さい。このことから、学習者は熟練者に比較し、素早く一気に引くことができていないことがわかる。また、熟練者は速度が上昇する前に「○」表示されているが、学習者は速度が上昇してから「○」が表示されている。このことから、熟練者は手元よりも腰が先に動いているが、学習者は手元が先に動いていることがわかる。

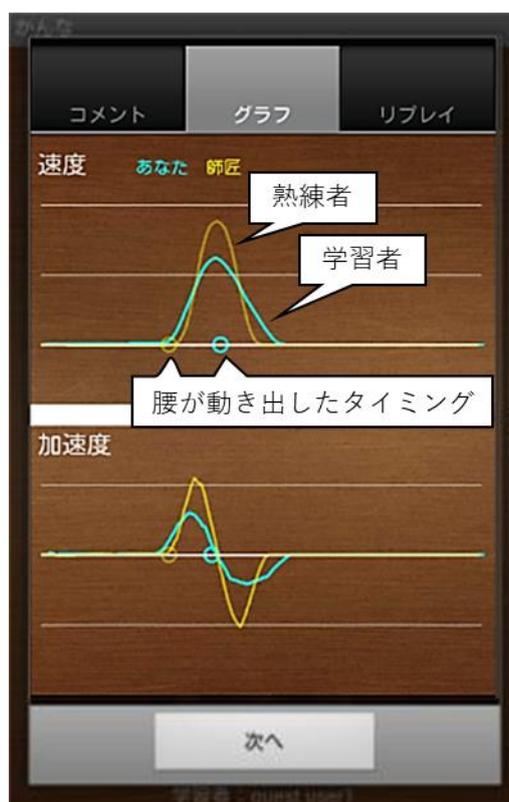


図 3-3 引きの速度および加速度のグラフ

「3.引きの速度を再現するアニメーション」(図 3-4)では、学習者の練習時の引きの速度と登録された熟練者の動作データの引きの速度がアニメーションで再現される。「リプレイ」のボタンをタップすると、2つのかんなのイラストが並んで速度を再現するため、学習者は速度の違いをより直観的に比較することができる。



図 3-4 引きの速度を再現するアニメーション

「①引きの速度」は、登録された熟練者の動作データの速度の最大値を 100 とした場合の割合を得点としている。学習者の速度がモデルの速度を超えた場合は全て 100 点となるようにした。次式は、学習者の速度の最大値を V_L 、モデルの速度の最大値を V_M とした際の、「①引きの速度」の得点 P_V を求める式である。例えば、モデルの引きの速度の最大値が 2.0 mm/sec で、学習者が 1.5 mm/sec だった場合、「①引きの速度」の得点は 75 点となる。

$$(V_L < V_M \text{ のとき}) \quad P_V = \frac{V_L}{V_M} \times 100$$

$$(V_L \geq V_M \text{ のとき}) \quad P_V = 100$$

「②引きの直進性」は、学習者のジャイロセンサの値の最大値を 100 とした場合の、登録された熟練者の動作データの値の割合を得点としている。学習者の値がモデルの値以下であった場合は全て 100 点となるようにした。次式は学習者のジャイロセンサの値を B_L 、モデルのジャイロセンサの値を B_M とした際の、「②引きの直進性」の得点 P_B を求める式である。例えば、モデルのジャイロセンサの値の最大値が 1.5 rad/sec² で、学習者が 2.0 rad/sec² だった場合、「②引きの直進性」の得点 P_B は 75 点となる。

$$(B_L > B_M \text{ のとき}) \quad P_B = \frac{B_M}{B_L} \times 100$$

$$(B_L \leq B_M \text{ のとき}) \quad P_B = 100$$

「③手元と腰の動き出しのタイミング」は 3 段階評価である。最も理想的なものが「A.腰が先に動いてから手元を引いている」動作である。その次に「B.腰も動いているが手元から引いている」動作であり、最も理想的でないものが「C.手元だけで引いている」動作である。それぞれの動作を表 3-4 にまとめた。

表 3-4 「③手元と腰の動き出しのタイミング」の観点における動作

動作	動き出しの順序
A.腰が先に動いてから手元を引いている	腰 → 手元
B.腰も動いているが手元から引いている	手元 → 腰
C.手元だけで引いている	手元のみ

| 第3章 運動技能のセンシングデータを基にフィードバックを提示する
 スマートフォンで動作する学習者用アプリケーションの開発と評価 |

学習者用アプリケーション使用時の流れについて、画面のフローを図 3-5 に示した。学習者用アプリケーションを起動すると、2 台のうち、どちらをかなな端末とし、どちらを腰端末とするかを選択するダイアログが表示される。かななと腰のそれぞれを選択すると、両端末が Bluetooth で接続される。接続が完了すると、腰端末には「腰付近のポケットに入れてください」のメッセージが表示される。この後、腰端末はセンシングのみの端末となり、学習者が操作する必要はないため、ポケットへ入れておく。かなな端末では Bluetooth 接続が完了すると、画面に「開始」のボタンが現れる。かなな掛けを開始できる状態で静止し、「開始」ボタンを押すと、「3・2・1・始め」と音声でかななを引くタイミングを指示される。合図に合わせてかなな掛けを行い、かなな掛けが終了すると、フィードバックが提示される。ここまで一連の練習のサイクルが終了する。その後、「開始」のボタンが表示される画面へ戻る。

熟練者の動作データの登録の流れも同様である。「開始」ボタンを押したあと、「3・2・1・始め」の合図に合わせてかなな掛けを行い、センシングされたデータに名前をつけて保存する。

熟練者の動作は複数保存することができ、練習前に使用したいデータを選択することができる。練習時にはその選択された熟練者のデータと比較され、フィードバックが提示される。

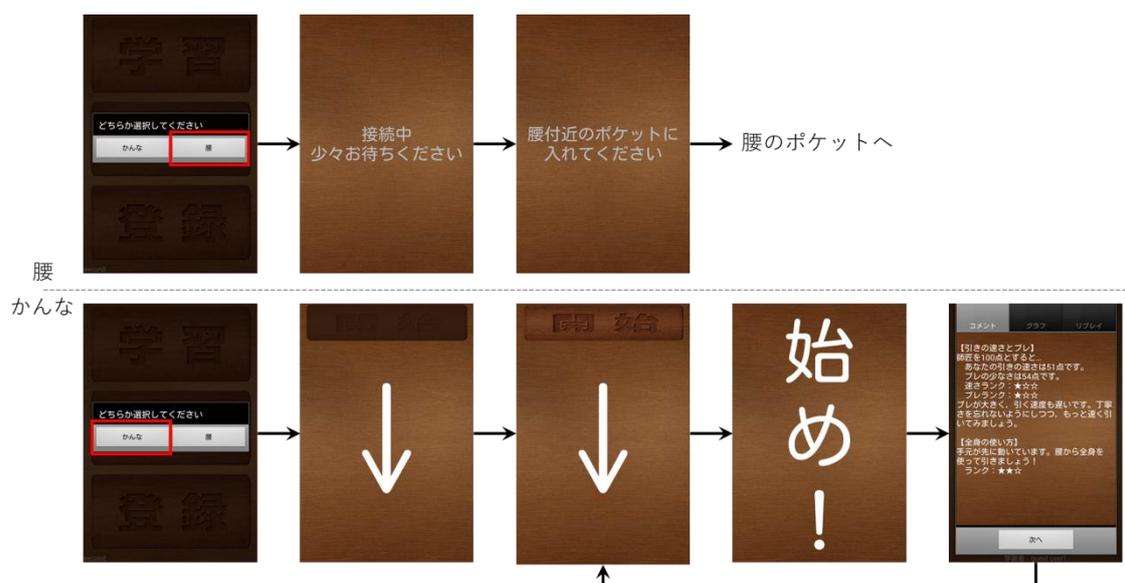


図 3-5 学習者用アプリケーションの画面のフロー

3.3. 専用かんなの開発

学習者用アプリケーションを使用してかんな掛けを行うために、スマートフォンを装着することができる専用かんなを開発した。

練習をしようとした際に、その準備に時間を必要としないことが望ましい。また、スマートフォンの画面が露出しているとかんなを握った際に誤操作を招く恐れがあるため、カバーで覆う必要がある。なお、覆うことにより画面の確認が困難になってしまわないような配慮も必要である。さらに、かんなを握ったときの実際の感覚を損なわないためには、かんなの元の形状や重量を可能な限り同じにするべきである。そこで、専用かんなの開発にあたっては、次の4点に留意した。

- (留意点1) スマートフォンの装着が簡単であること
- (留意点2) スマートフォンの誤操作を防ぐこと
- (留意点3) かんなの元の形状を可能な限り保つこと
- (留意点4) かんなの元の重量を可能な限り保つこと

開発した専用かんなを図 3-6 に示す。基となるかんなは275×80×36mmのものである。かんな台（木製部分）の中央部から台尻の部分を、フライス盤や鑿を用いて加工した。加工後のかんな台の寸法を図 3-7 に示す。



図 3-6 開発した専用かんな

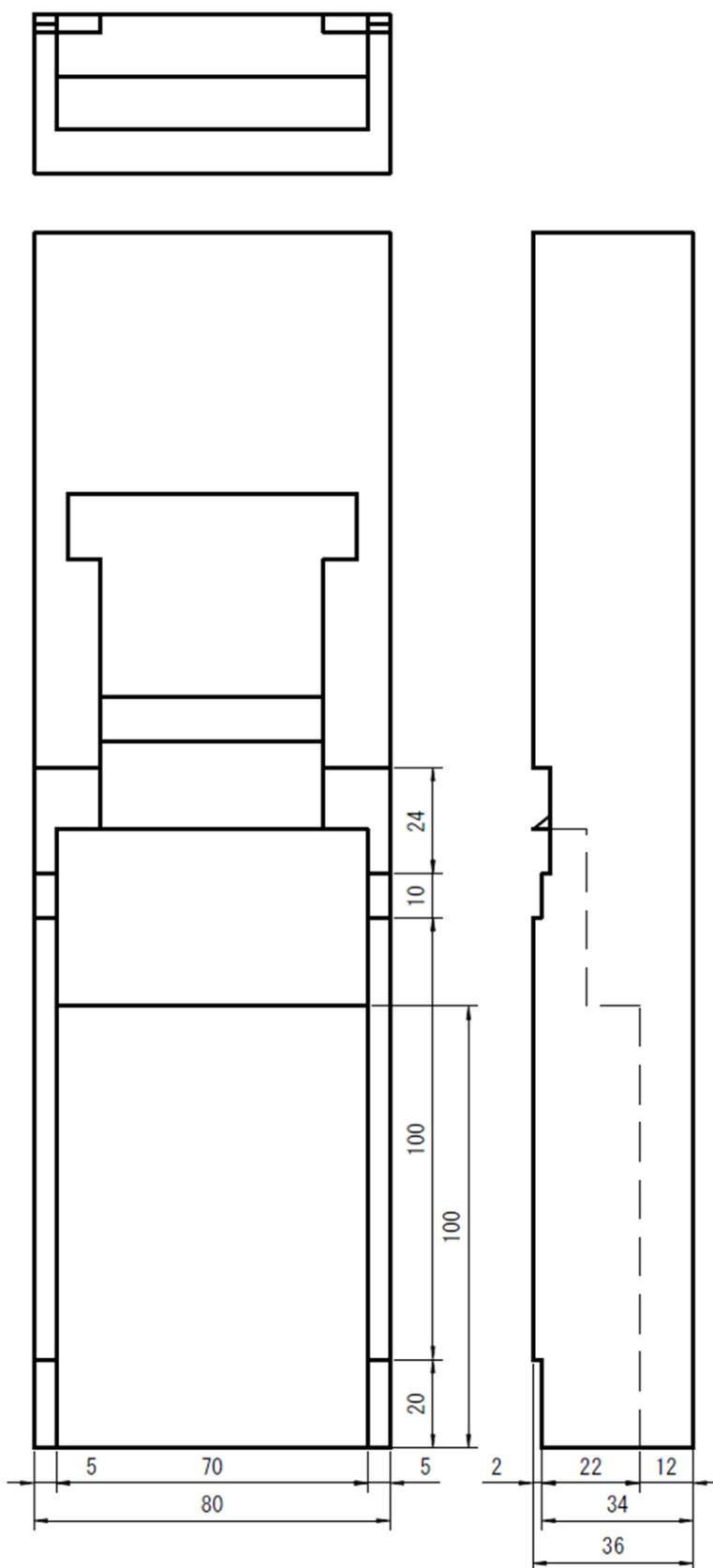


図 3-7 加工後のかな台の寸法

図 3-6 の白色部分は 3D プリンタを用いて製作した端末固定部（図 3-8）である。ABS 樹脂を使用した。図 3-8 の A 部と B 部はばねでつながっている。A 部はかんな台にねじ止めしてあり、B 部がスライドすることで、簡単にスマートフォンを装着することができる（留意点 1）。この端末固定部の A および B のそれぞれの図面を図 3-9 および図 3-10 に示す。誤操作を防ぐために透明の亚克力板でカバーできるようにした（留意点 2）。カバー（図 3-11）は透明であるため、スマートフォンの画面は常に確認できる。カバーは、3D プリンタで製作した小型の蝶番（図 3-12）でかんな台に接合されており、持ち上げることで扉のように開閉することができる。蝶番の接合部分は蝶番の寸法に合わせてかんな台を凹加工し、埋め込んでいるため、カバーを閉めたときのかんなの形状は、加工前のものと同じである（留意点 3）。

表 3-5 は加工前のかんなと、加工後のかんなの重量の比較である。加工前が 1104 g であるのに対し、加工後（スマートフォン装着時）は 1060 g である。44 g の差があるが、かんな掛けを行う上では影響のない範囲の差であると判断した（留意点 4）。

表 3-6 は専用かんなの製作に必要な部品の一覧である。

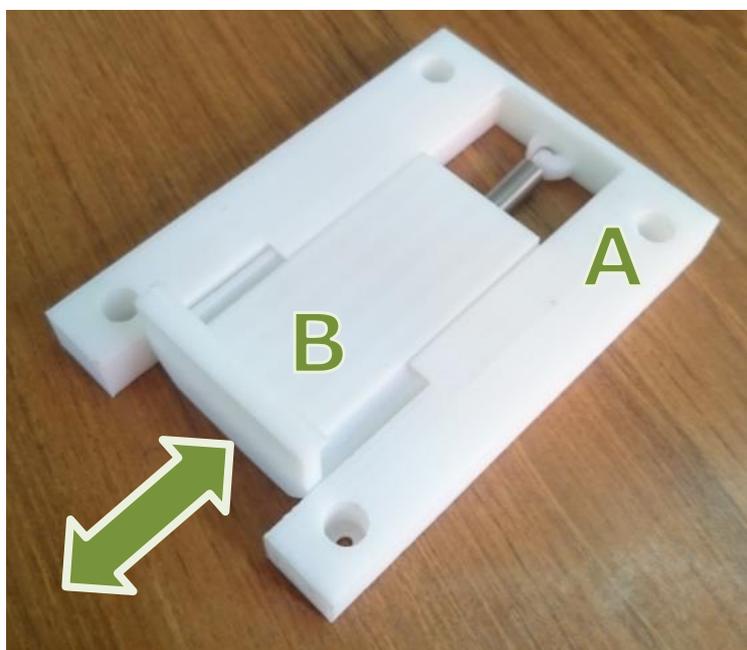


図 3-8 専用かんなの端末固定部分

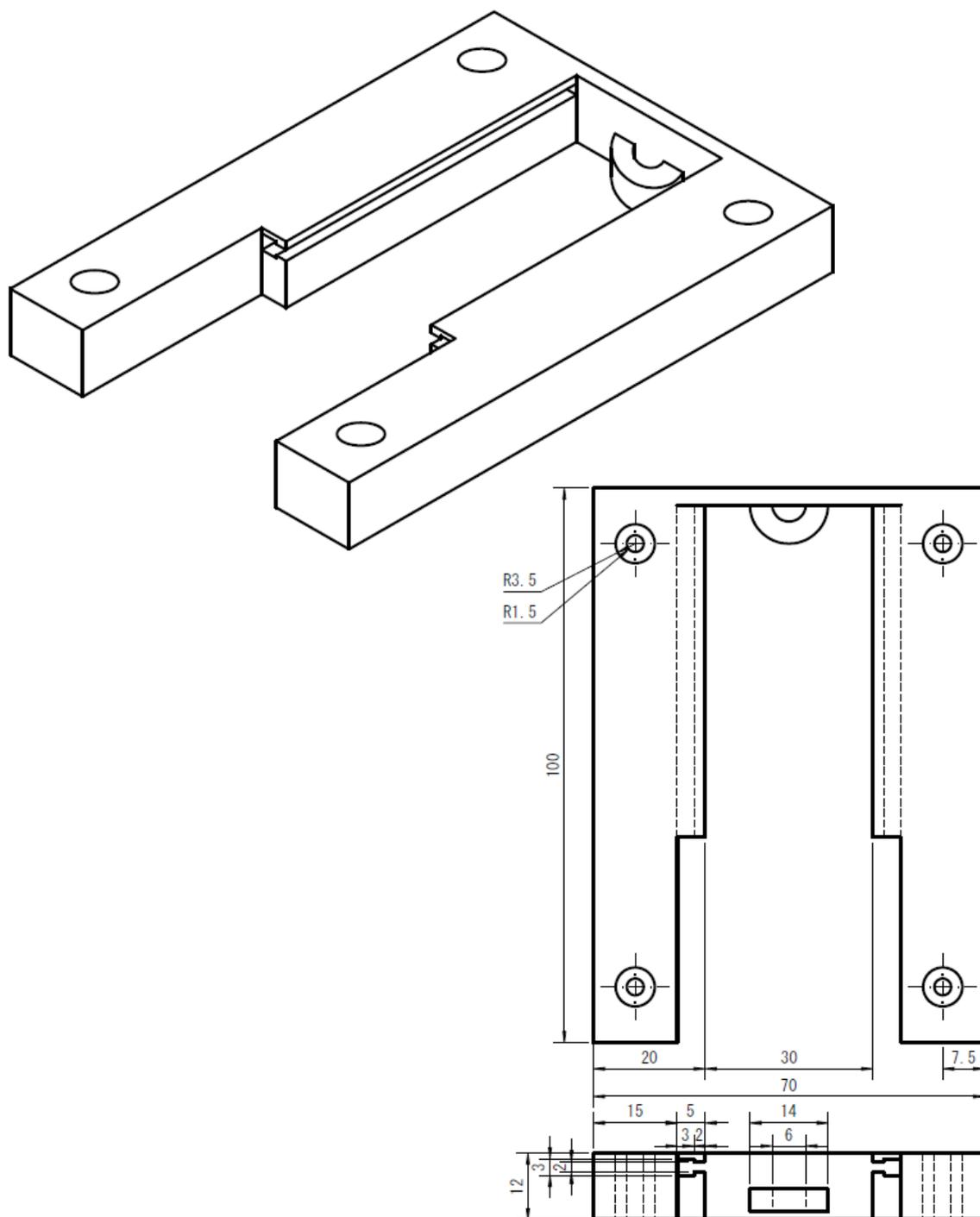


図 3-9 端末固定部 (A) の図面

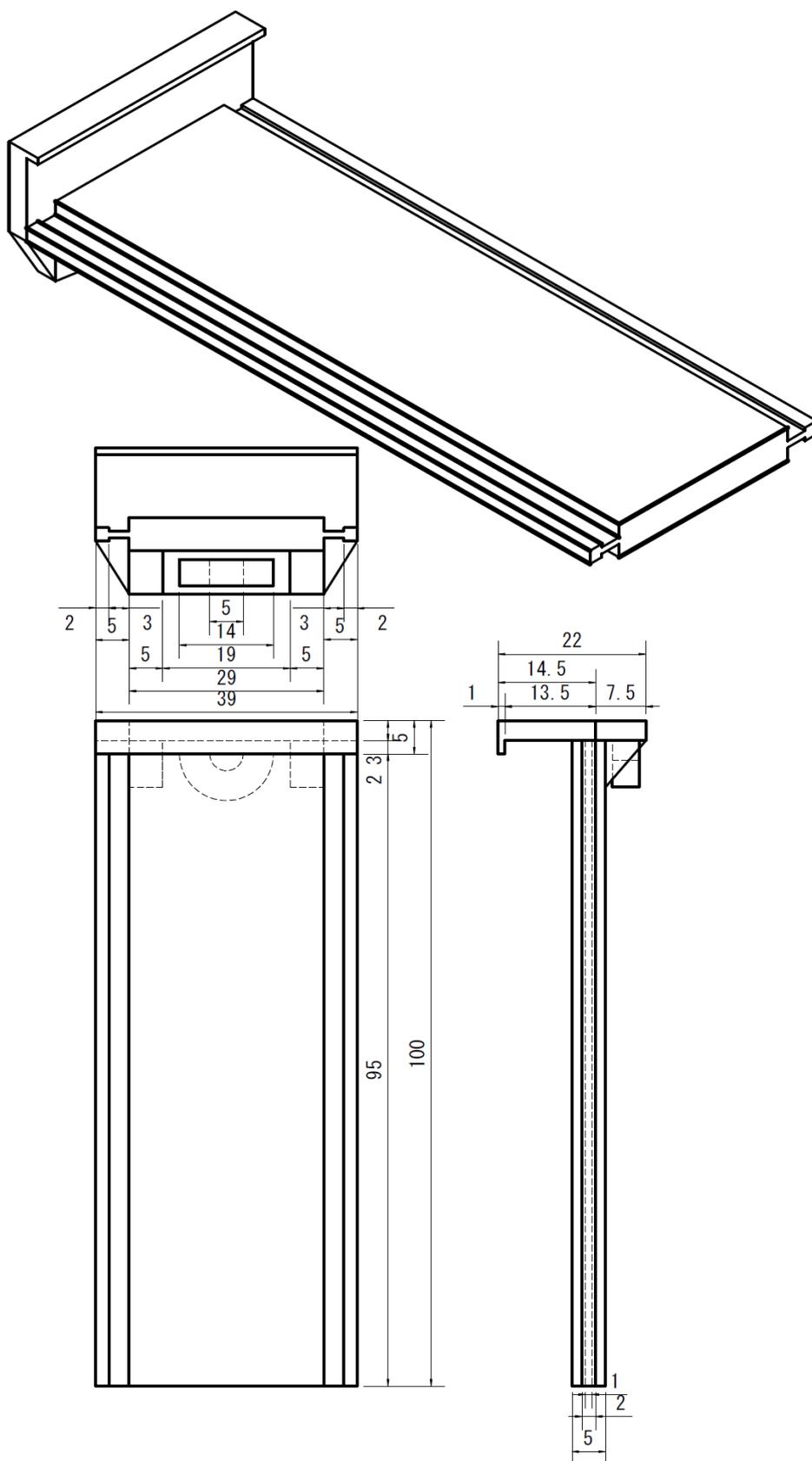


図 3-10 端末固定部 (B) の図面

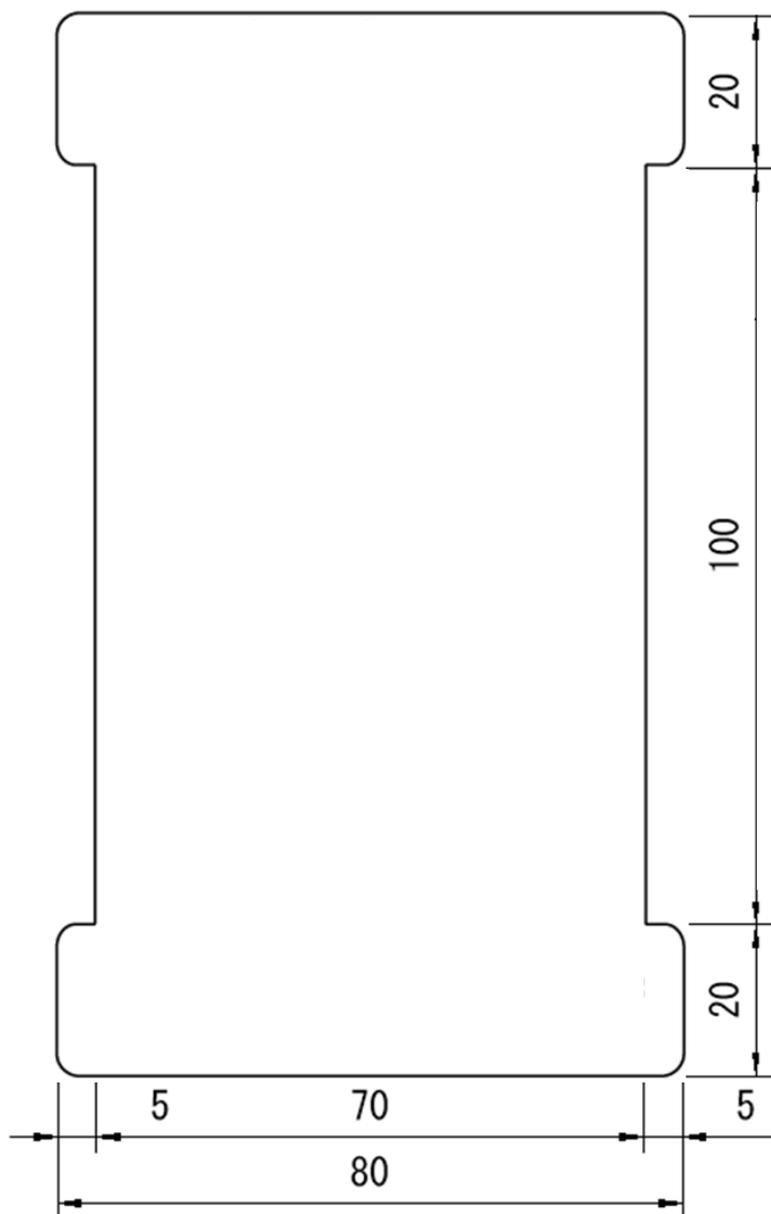


図 3-11 カバーの図面 (厚さは 2 mm)

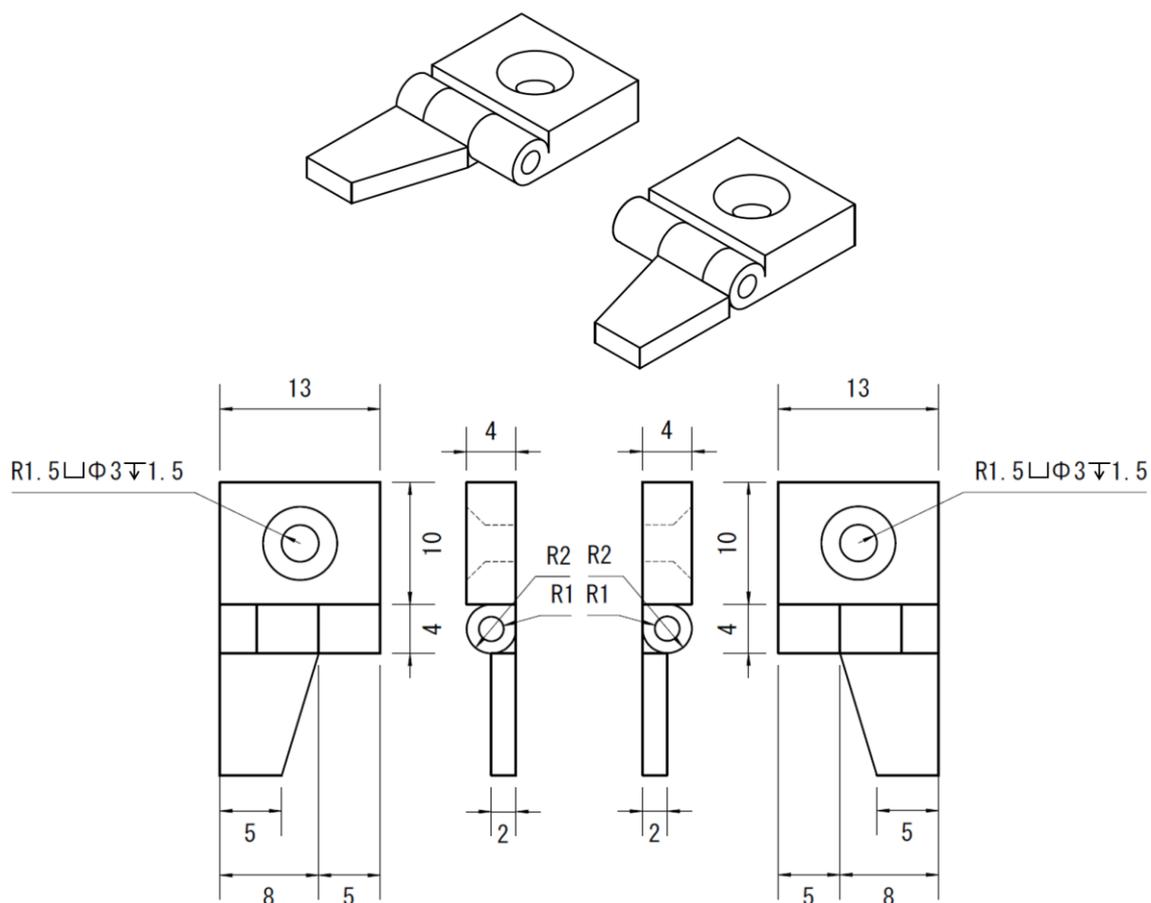


図 3-12 蝶番の図面

表 3-5 かんなの重量の比較

かんな	重量[g]
加工前	1104
加工後(スマートフォン未装着時)	930
加工後(スマートフォン装着時)	1060

表 3-6 専用かんなの製作に必要な部品の一覧

	数量	寸法	
寸六かんな	1	275×80×36 mm	-
アタッチメント	A, B 各 1	図 3-9, 図 3-10 参照	3D プリンタで製作
蝶番	左右各 1	図 3-12 参照	3D プリンタで製作
カバー	1	図 3-11 参照	レーザーカッターで製作
ばね	1	0.65×6.3×52 mm	寸法は線径×外径×長さ
タッピングネジ (鍋)	4	φ3×10 mm	-
タッピングネジ (皿)	2	φ3×5 mm	-
接着剤	-	-	-

3.4. 学習者用アプリケーションの評価

3.4.1. 方法

学習者用アプリケーションの効果を評価するための実験を行った。実験参加者は H 中学校第 2 学年の生徒 14 名 (男 9 名, 女 5 名) であった。場所は同中学校の技術室であった。使用した機材等は, 専用かんな, スマートフォン, 削るための棒材である。棒材は長さ 400 mm のものを使用した。実験は 2016 年 11 月 29 日に行われた。

モデルとなる熟練者のかんな掛けのデータとして, かんな掛け経験 14 年を有する M 大学の技官の動作をセンシングし, データを事前に登録した。この動作データでは, 「①引きの速度」の最大値が 1.78 m/sec であった。田中ほか (1990, 1991) の先行研究では, 熟練者の引きの速度の最大値は約 1.7 m/sec であり, 本研究の 1.78 m/sec とほぼ同等であるため, 熟練者のかんな掛けとして妥当であると判断した。なお, このかんな掛けのデータでは, 「②引きの直進性」が 1.0 rad/sec² であり, 「③手元と腰の動き出しのタイミング」においては, 手元より腰の方が 0.1 秒早く動き出していた。

実験の流れについて, はじめに同中学校の技術・家庭科技術分野の教師から, 動画を用いて, 学習者用アプリケーションの操作方法の説明が行われた。次に, 学習者用アプリケーションの操作方法の確認も兼ねながら, かんな掛けの演示が教師により行われた。その後, 生徒は二人一組になり, 学習者用アプリケーションを用いて練習を行った。練習はおよそ 30 分間, 教師からの終了の合図があるまで繰り返された。

図 3-13 は学習者が練習をしている様子である。作業台に備え付けられた万力に棒材を固定して、専用かんなを用いて削っている。ポケットには腰端末が入れている。

なお、学習者用アプリケーションの純粹な効果を評価するためには、使用の有無以外の条件を統制し、比較を行うことが望まれる。しかし、本実験においては中学生を対象としており、教育の平等性の観点から研究の限界と判断し、統制条件を設定せずに実験を行った。



図 3-13 学習者が練習している様子

3.4.2. 結果

各生徒の最初の1回を開始時、最後の1回を終了時とし、開始時と終了時における3観点それぞれの得点の比較を行った。生徒の平均練習回数は13.6回であった。

本来、3観点を総合的に評価し、協応動作をとれているかも評価すべきであると考えられる。しかし、学習者用アプリケーションでは、「①引きの速度」および「②引きの直進性」については、熟練者を100としたときの割合を得点とするという、同一の手法で得点を算出しているが、「③手元と腰の動き出しのタイミング」については、「A.腰が先に動いてから手元を引いている」、「B.腰も動いているが手元から引いている」、「C.手元だけで引いている」の3段階での評価であり、3観点の結果を統合することが困難であった。また、かんな掛けに関する先行研究においても、熟練者と未熟練者の相違点を挙げているものが見られるが（例えば、田中・安孫子 1990, 安孫子ほか 2005）、そこで挙げられた複数の相違点がどのように影響しあっているかを述べたものは見当たらない。以上の理由より、3観点の評価を統合する妥当な手法が考えられなかったため、本研究においては3観点を個別に扱うこととした。

分析の方法について、「①引きの速度」および「②引きの直進性」では線形混合モデル（Linear Mixed Model : LMM）によって、「③手元と腰の動き出しのタイミング」では一般化線形混合モデル（Generalized Linear Mixed Model : GLMM）のロジット混合分析によって分析した。分析にあたっては、統計パッケージ R（ver. 3.4.3 ; <https://www.r-project.org/>）を用いた。LMMによる分析では、lme4 パッケージの lmer 関数を、GLMMによる分析では同パッケージの glmer 関数を用いた。ランダム要因に関して、実験参加者について切片および開始時と終了時の条件の傾きを設定したモデルを最大モデルとし、より簡略なモデルとして切片のみを設定したモデルを設定した。尤度比検定の結果、両モデルに有意な差はみられなかった場合には、切片のみを設定したモデルを採用した（結果的に、以降に述べる分析すべてにおいて、切片のみを設定したモデルが採用された）。

【①引きの速度】

計測時点（開始時／終了時）を固定要因とした。各生徒によって学習回数が異なることによる影響を統制するために、練習回数も固定要因に加えた。また、ランダム要因に関して、実験参加者について切片のみを設定したモデルを採用した。分析の結果（図 3-14、表 3-7），計測時点の主効果が有意傾向（ $p=0.068$ ）であり，開始時（ $M=70.1$, $SD=24.7$ ）に比べ，終了時（ $M=84.9$, $SD=15.7$ ）が高いことが確認された。練習回数の影響は見られなかった。

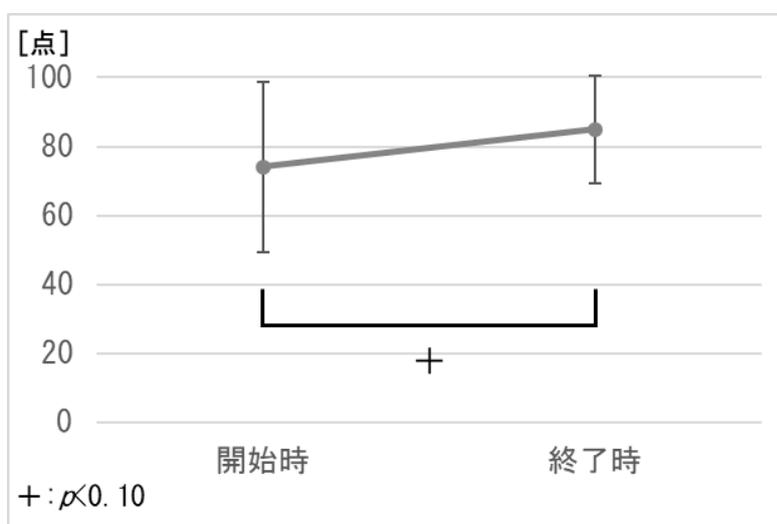


図 3-14 「①引きの速度」の結果

表 3-7 「①引きの速度」の結果

	推定値	標準誤差	t 値	p	
(切片)	76.63	9.69	7.91	<0.001	***
計測時点	10.86	5.47	1.98	0.069	+
練習回数	0.21	0.61	0.35	0.736	

** $p < 0.005$, + $p < 0.10$

【②引きの直進性】

①の分析と同様の固定要因およびランダム要因を設定したモデルを採用した。分析の結果（図 3-15, 表 3-8），開始時／終了時の主効果が有意傾向（ $p=0.052$ ）であり，練習開始時（ $M=53.1$, $SD=31.0$ ）に比べ，終了時（ $M=70.8$, $SD=27.4$ ）が高いことが確認された。練習回数の影響は見られなかった。

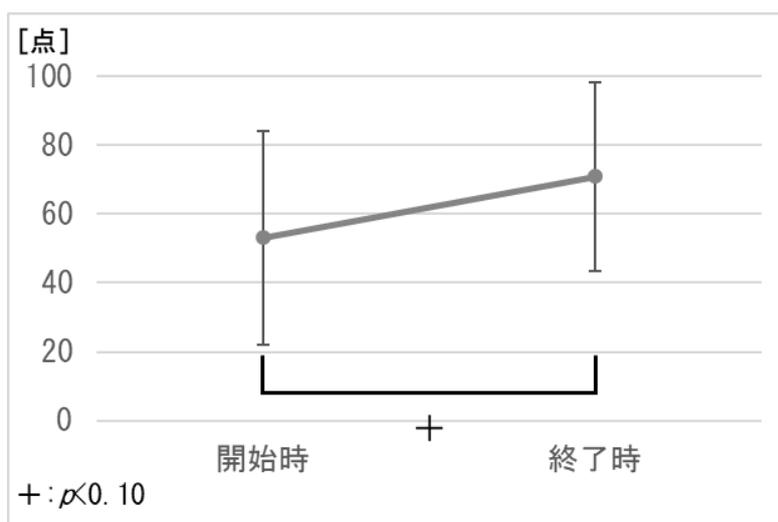


図 3-15 「②引きの直進性」の結果

表 3-8 「②引きの直進性」の結果

	推定値	標準誤差	t 値	p	
(切片)	54.05	13.12	4.12	0.001	**
計測時点	17.79	8.32	2.14	0.052	+
練習回数	0.59	0.82	0.71	0.491	

** $p<0.005$, + $p<0.10$

【③手元と腰の動き出しのタイミング】

理想的な動作である「A.腰が先に動いてから手元を引いている」を「1」、不完全な動作である「B.腰も動いているが手元から引いている」および「C.手元だけで引いている」を「0」とし、ロジット混合モデルによって分析した。①および②の分析と同様の固定要因およびランダム要因を設定したモデルを採用した。分析の結果（図 3-16, 表 3-9），計測時点の主効果は有意ではなく，上達は確認されなかった。また，練習回数の影響も有意ではなかった。

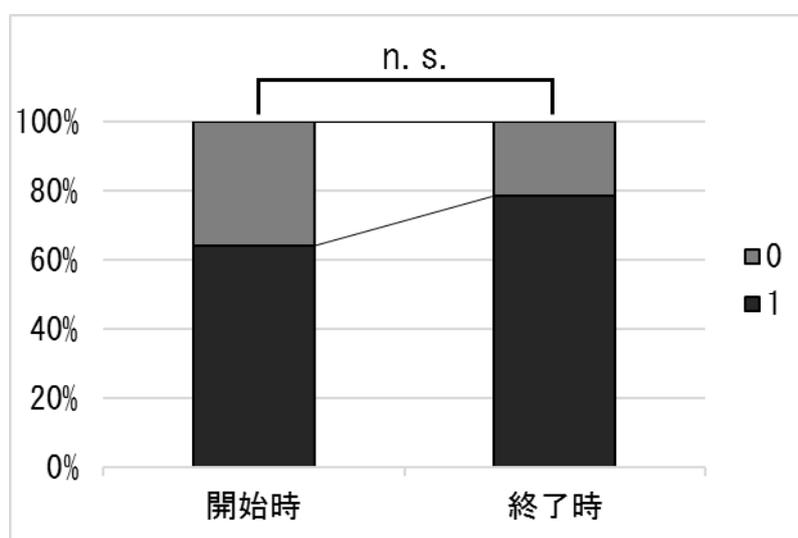


図 3-16 「③手元と腰の動き出しのタイミング」の結果

表 3-9 「③手元と腰の動き出しのタイミング」の結果

	推定値	標準誤差	z 値	p
(切片)	0.71	0.83	0.86	0.390
計測時点	0.71	0.86	0.83	0.406
練習回数	0.02	0.05	0.32	0.751

3.4.3. 考察

「①引きの速度」および「②引きの直進性」の2観点の上達において、有意傾向が確認された。速く引くために力を込めることにより、引きの直進性は低下することが考えられるが、その両観点の得点が共に向上したことから、学習者はフィードバックを受けて両観点を意識し、速く引きつつも丁寧に引くよう、自身の動作を制御することができたものと考察できる。本実験では統制条件を設定しておらず、学習者用アプリケーションによるフィードバックの効果であったと断定することはできないが、効果が示唆される結果であったといえる。

一方、「③手元と腰の動き出しのタイミング」においては、有意な上達は確認されなかった。練習開始時の時点で半数以上の学習者が、理想的な動作である「A.腰が先に動いてから手元を引いている」であったため上達が確認されにくかったことは考えられるが、学習者用アプリケーションでは3段階の評価にとどまっているため、手元と腰のどちらがどの程度早く動き出していたかといった詳細なフィードバックを提示するなど、学習者用アプリケーションに改良の余地があることが見出された。評価している観点のうち、改善が必要な観点を学習者により強く意識させるために、向上している観点よりも優先して画面上部に表示させるなどの方法も考えられる。

3.5. まとめと課題

本章では、かんな掛けを対象とし、身近に普及しているスマートフォンを用いて、学習者の運動技能をセンシングし、取得したデータを基にフィードバックを提示する学習者用アプリケーションの開発について述べた。また、学習者用アプリケーションを用いて学習者のかんな掛けをセンシングするために、スマートフォンを装着することができる専用のかんなの開発について述べた。そして、これらを用いて運動技能の学習における連合段階および自動化段階の学習支援が可能か評価する実験を行った。

その結果、評価している3観点のうち、「①引きの速度」および「②引きの直進性」の2観点の上達において有意傾向が確認された。スマートフォンという身近なデバイスを用いて、従来のセンシングの手法よりも低いコストで、かつ簡便に運動技能をセンシングして、取得したデータを基にフィードバックを提示することができた。また、これにより、運動技能の学習における連合段階および自動化段階の学習支援が可能であることが示唆された。

本章で得られた知見はかんな掛けを対象としたものであり、対象とする運動技能が変わ

れば、評価する観点やその数は異なり、そのまま応用することは難しい。しかし、対象とする運動技能の然るべき観点において、スマートフォンを用いて運動技能をセンシングし、熟練者の動作と比較した結果をフィードバックとして提示する手法は、他の運動技能の学習支援を検討する際に応用可能である。

学習者用アプリケーションを、対面による学習と e-Learning による学習で活用する場合、それぞれ次に示す点が検討されるべきであることが課題といえる。

対面による学習で活用する場合、学習者が学習者用アプリケーションを用いることで個々の学習者の支援は実現することができると考えられる。さらに、対面による学習ではその場に指導者が存在する。中山・東原（1986）や石塚ほか（2007）の知見に基づけば、取得したセンシングデータを収集して指導者に提示することにより指導を効率化でき、学習者がさらなる指導を受けることができるようになることを期待できる。これについて、第4章で述べる。

e-Learning による学習で活用する場合、e-Learning の利点の1つである「いつでもどこでも学習できる」ことは重要であるといえる。従来のセンシングの手法のような専用の装置がある環境でしか学習できない場合、簡便であるとはいえない。そこで、家庭学習などの、専用かんなや削るための木材、削り台などの実践的な学習環境が整っていない模擬的な学習環境においても、同様の効果を得られるか検討する必要がある。これについて、第5章で述べる。

第4章 運動技能のセンシングデータを基に指導参考情報を提示する 教師用システムの開発と評価

前章では、学習者用アプリケーションの開発および評価について述べた。本章では、学習者用アプリケーションを用いた、運動技能の学習における連合段階および自動化段階の、対面による学習における学習支援について検討した。対面による学習では、その場に指導者が存在するため、その指導を効率化することにより、学習者をさらに支援することが可能になると期待できる。そこで、学習者用アプリケーションを拡張し、取得したセンシングデータを基に指導参考情報を提示する教師用システムを開発し、授業実践を通して評価を行った。

4.1. 目的

小学校、中学校および高等学校の学習指導要領では、個に応じた指導を充実させることが求められている。しかし、現状では、少人数学級の更なる推進が求められている（文部科学省 2012）ことからわかるように、1学級の40人近い学習者が一人の教師から受けることができる指導には限界があると考えられる。一人一人の学習者全員が十分な指導を受けるだけの時間を確保できれば良いが、かんな掛けが扱われる中学校技術・家庭科技術分野においては特に、授業時数の不足が長年叫ばれている（河合ほか 2014）。

このような現状に対して、運動技能の学習における連合段階および自動化段階の学習において、学習者用アプリケーションを活用することで個々がフィードバックを受けながら学習できることには価値があるといえる。これに加え、中山・東原(1986)や石塚ほか(2007)のように、取得したデータに基づいて教師による指導を効率化することができれば、学習者がさらなる指導を受けることができるようになると考えられる。

本章では、学習者用アプリケーションを拡張し、取得したセンシングデータを基に指導参考情報を提示する教師用システム（以下、教師用システム）を開発し、授業実践を通して、運動技能の学習における連合段階および自動化段階の、対面による学習における学習支援が可能か評価することを目的とした。

4.2. 教師用システムの概要

システム全体の構成を図 4-1 に示す。教師用システムは、「サーバ部」および「指導参考情報提示部」から構成される。学習者用アプリケーションおよび専用かんな（以下、学習者用端末部）と連携し、センシングされたデータがサーバ部へ蓄積され、そのデータを基に、教師に指導参考情報を提示する。

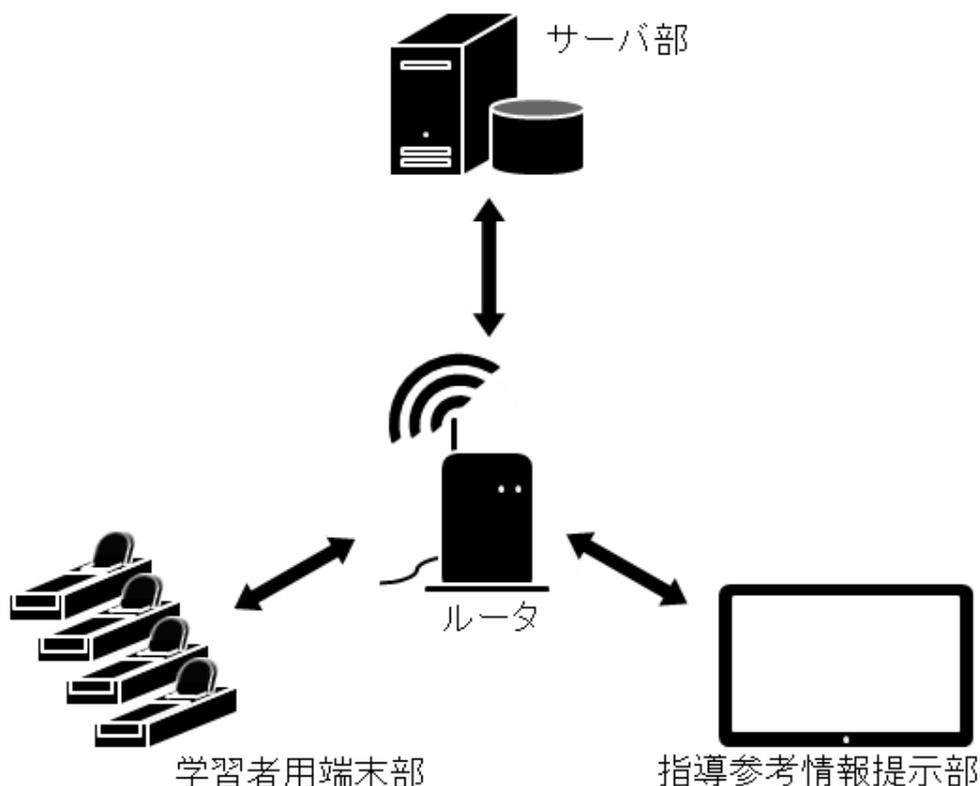


図 4-1 システム全体の構成

「サーバ部」は1台のコンピュータであり、Webサーバとデータベースサーバの両方の役割を果たしている。コンピュータのCPUはCore i5 2.50 GHz、RAMは8 GBであり、Windows 8.1上でApacheが動作している。データベースにはSQLiteを用いた。開発言語にはPHP、JavaScriptおよびHTMLを用いた。学習者用端末部との通信は、PHPにより組まれたサーバサイドのプログラムと、HTTP通信を行うことでデータを送受信している。学習者用端末部がサーバ部へデータを送信する場合には、PHPプログラムへGET送信でデータを送っている。学習者用端末がサーバ部からデータを受信する場合には、PHPプログラムにより成形されるテキストデータを受け取っている。次節で述べるグラフの描画には、HTML5.JP (<http://www.html5.jp/library/>) の、レーダーチャートおよび折

れ線グラフのライブラリを使用した。

データベースの ER 図を図 4-2 に示す。データベースには、学習者用端末部を用いて学習者が練習を行った直後に送信される練習情報を受け取り、蓄積している。後述する指導参考情報提示部では、ここに蓄積されたデータを基に情報を提示している。また、事前に学年、学級、氏名等の学習者情報を登録することができる。図 4-3 は学習者情報を登録する画面である。基本的には学習者情報が記された CSV ファイルから一括して登録することができる。必要に応じて個別に登録することも可能である。

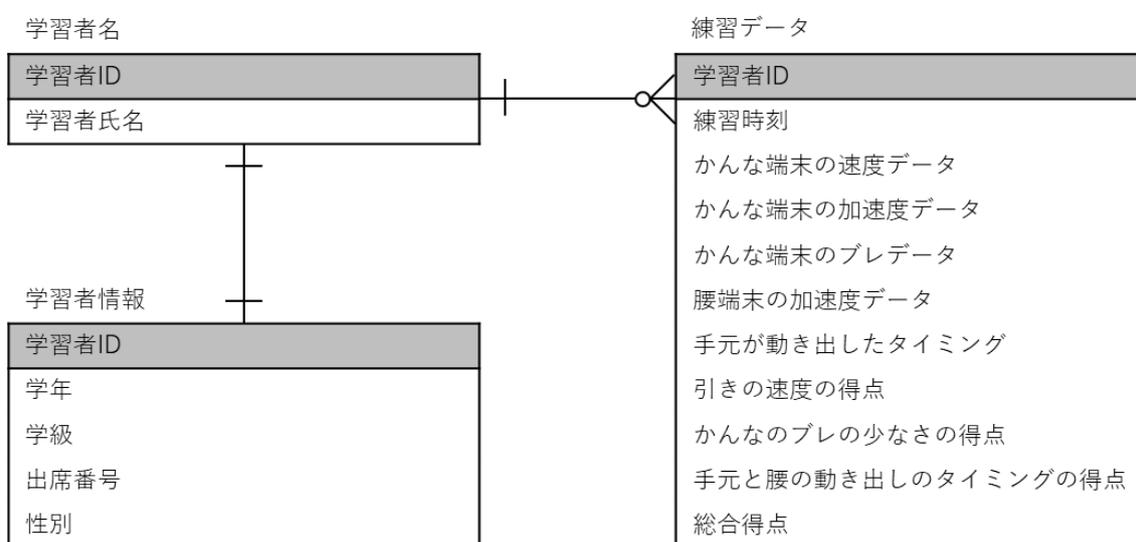


図 4-2 データベースの ER 図

生徒情報確認	データベース内の生徒の情報を確認できます。
生徒情報追加	データベースに生徒を1人ずつ追加できます。
生徒情報一括追加	CSVファイルから複数の生徒を一括で追加できます。
生徒情報更新	CSVファイルから生徒情報を更新します。 (既存のデータは上書きされます)
CSVフォーマット	一括登録用のCSVのフォーマットをダウンロードできます。

図 4-3 学習者情報を登録する画面

第4章 運動技能のセンシングデータを基に指導参考情報を提示する
教師用システムの開発と評価

「指導参考情報提示部」は、教師が学習者の練習履歴へアクセスし、指導の参考になる情報を得るためのものである。コンピュータやタブレット PC などの情報端末の Web ブラウザからアクセスして使用する。サーバ部から提示される指導参考情報は「1.グラフ表示」, 「2.リスト表示」, 「3.上達履歴表示」の3つがある。

「1.グラフ表示 (図 4-4)」では、学習者用アプリケーションに表示されるものと同様の速度や加速度のグラフや、3 観点の得点のバランスを確認できるレーダーチャートが表示される。速度や加速度のグラフでは、ブレの大きさの変化も重ね合わせて表示されるため、削る動作の中で、いつかんながぶれているのかを確認することも可能である。データが多数存在する場合にも必要なデータのみを確認できるよう、学級別の表示や、期間を指定した絞り込み、特定の学習者のみの表示なども行うことができる。

なお、評価では観点を個別に扱っているが、理想的には一部の観点到に偏ることなく総合的に上達することが望ましいため、3 観点の結果から概算した「総合得点」も指導参考情報提示部や学習者用端末部で表示している。

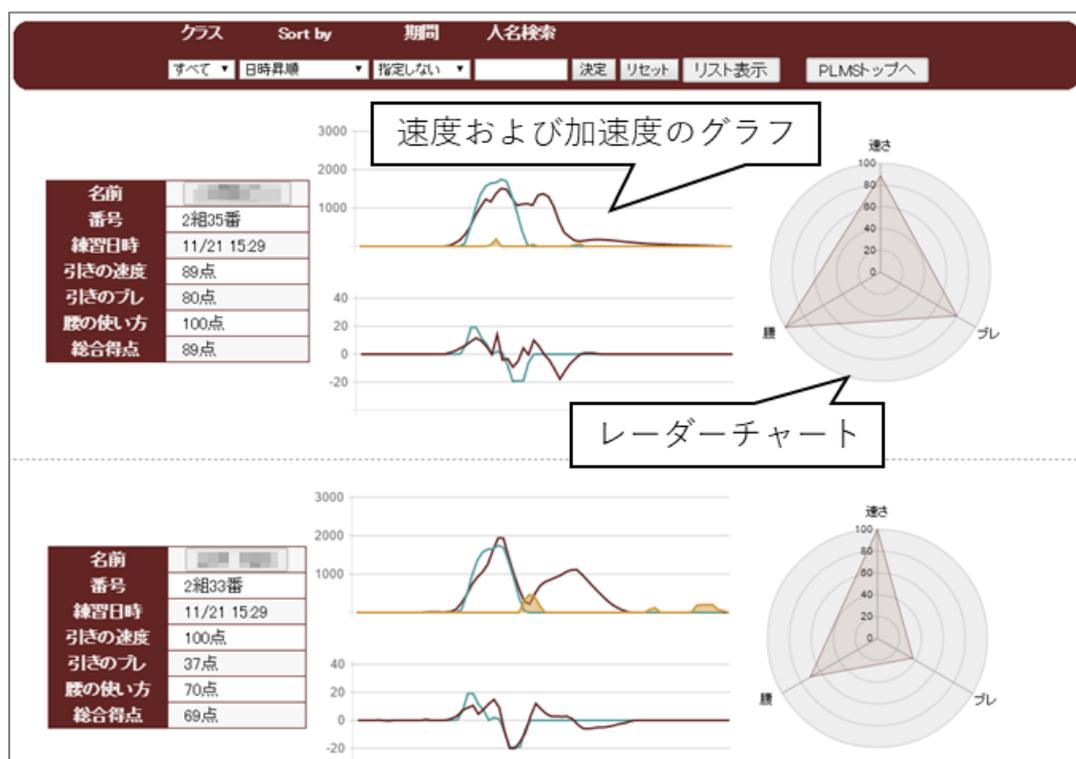


図 4-4 グラフ表示

第4章 運動技能のセンシングデータを基に指導参考情報を提示する
教師用システムの開発と評価

「2.リスト表示 (図 4-5)」では、学習者全員の練習データを表形式で一覧表示する。「1.グラフ表示」のように詳細な結果は表示されないが、「★」の数で表される各観点の得点のランクが一目で確認できるように表示される。また、一定の得点に満たない学習者が赤く強調表示される。さらに「1.グラフ表示」と同様の学級別の表示や、得点の降順・昇順のソートも可能にし、指導者が、指導が必要な学習者を効率的に見つけることができるよう設計した。

クラス	番号	名前	得点	引込さ	フリの少なさ	腰の使い方	日時
2	4		61点	★★★	★★	★	11/21 15:30
2	8		95点	★★★	★★★	★★★	11/21 15:30
2	10		80点	★★★	★★★	★★	11/21 15:30
2	1		69点	★★★	★	★★★	11/21 15:30
2	2		100点	★★★	★★★	★★★	11/21 15:30
2	35		85点	★★★	★	★★★	11/21 15:30
2	4		62点	★★★	★★	★	11/21 15:29
2	35		89点	★★★	★★	★★★	11/21 15:29
2	33		69点	★★★	★	★★	11/21 15:29
2	26		72点	★★	★	★★★	11/21 15:29
2	22		31点	★	★★	★	11/21 15:29
2	5		61点	★★★	★	★	11/21 15:29
2	17		83点	★	★★★	★★★	11/21 15:29
2	21		100点	★★★	★★★	★★★	11/21 15:29
2	8		100点	★★★	★★★	★★★	11/21 15:29
2	3		95点	★★★	★★★	★★★	11/21 15:29
2	13		97点	★★★	★★★	★★★	11/21 15:29
2	1		90点	★★★	★★	★★★	11/21 15:29
2	4		69点	★★★	★★★	★	11/21 15:29
2	23		73点	★★	★	★★★	11/21 15:29

図 4-5 リスト表示

「3.上達履歴表示（図 4-6）」では、各学習者の観点ごとの得点の遷移が表示される。指導者はそれぞれの学習者がどの点で伸び悩んでいるのかを確認することができる。また、この「3.上達履歴表示」は学習者用アプリケーション内にも表示される（図 4-7）ため、学習者自身も得点の推移を確認することができる。

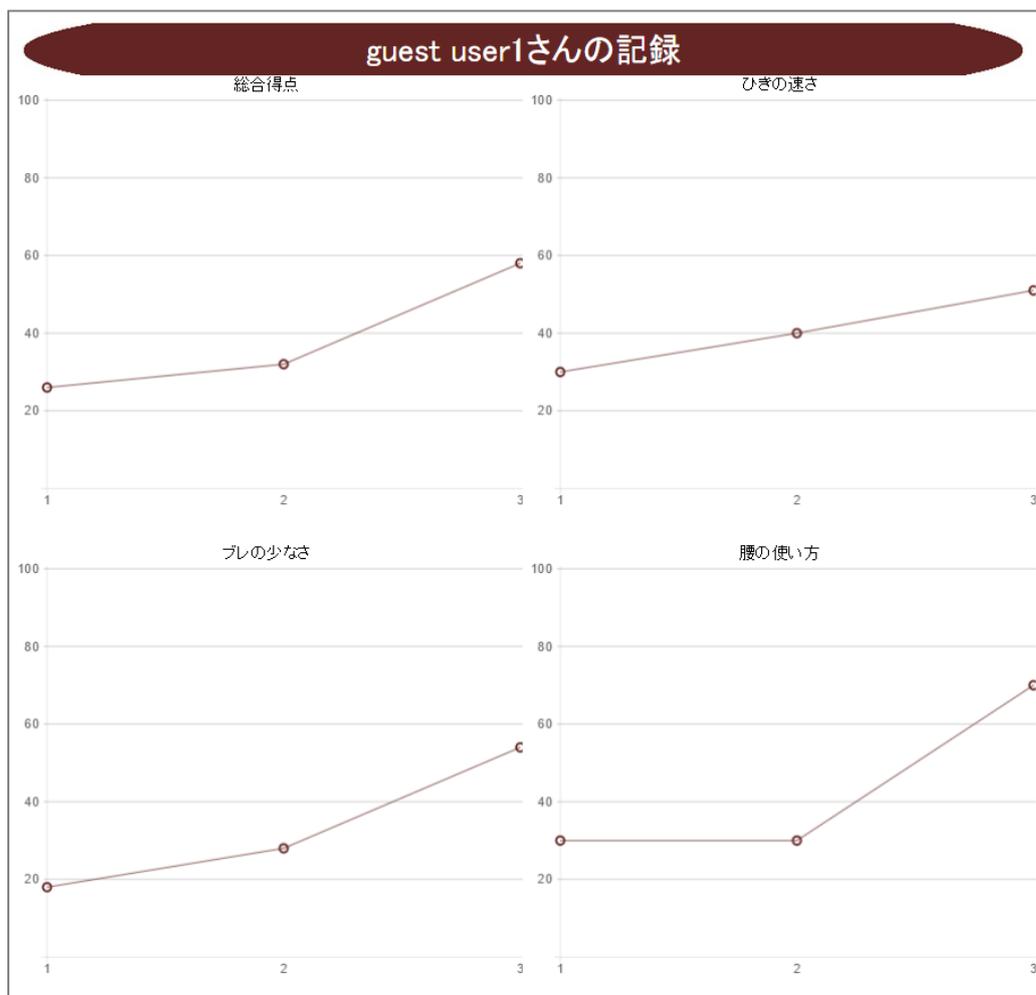


図 4-6 上達履歴表示

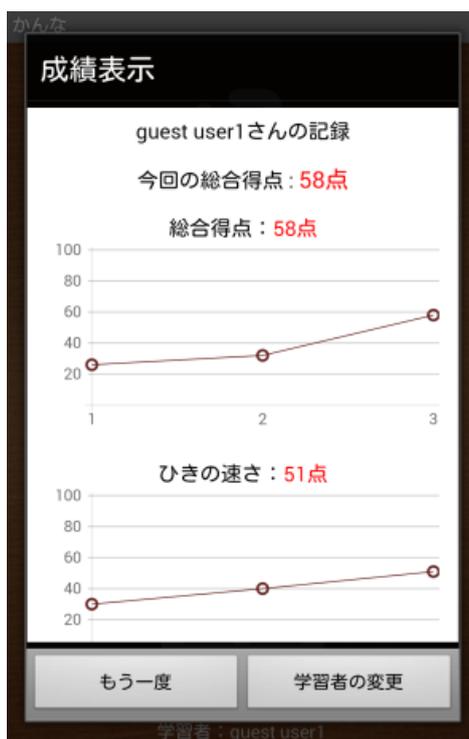


図 4-7 学習者用端末部に表示される上達履歴表示

「学習者用端末部」は前章で述べたものを3点拡張した。

1点目は、学習者の選択である。サーバ部にデータを蓄積する際、どのデータがどの学習者によるデータであるのかを紐づけるために、練習前に学習者を選択する操作を求めよう修正を加えた。具体的には、かんな端末と腰端末のそれぞれを選択し、両端末がBluetoothで接続された後、学級の選択および出席番号の選択を行う画面が表示されるようにした。また、フィードバックが提示された後、他の学習者と交代できるようにするため、再度練習をするための「もう一度」のボタンに加え、「学習者の変更」のボタンを表示させた。「学習者の変更」のボタンをタップすると、出席番号を選択する画面が表示され、出席番号を選択すると、学習者が切り替わった状態で練習を行うことができるようになる。

2点目は、登録された熟練者の動作データを、サーバ部からダウンロードする機能である。学級内の学習者全員が同一のモデルに基づいて練習することができるよう、学習者用アプリケーション起動時に、サーバ部内に保存されている熟練者のデータをダウンロードするよう修正を加えた。

3点目は、上記の図4-7に示した上達履歴表示の表示である。前章の図3-14から図3-16に示した3つのフィードバックが提示された後、上達履歴表示が表示されるよう修正を加

えた。

以上の修正後の、学習者用アプリケーションの使用時の流れを図 4-8 に示す。網掛け部分は、前章で述べた修正前の学習者用アプリケーションの使用時の流れからの変更部分である。



図 4-8 教師用システムと連携時の学習者用アプリケーションの画面のフロー

4.3. 教師用システムの評価

4.3.1. 方法

教師用システムを授業において評価するための実験を行った。実施日は、2014年11月11日、17日、21日であった。実験参加者はS中学校の技術・家庭科技術分野の教師1名および第1学年の3学級の生徒101名であった。教師が授業者となり、3学級それぞれで授業を行った。

場所は同校の技術室であった。準備のしやすさ等の観点から、無線LANルータを1台設置し、サーバ部と指導参考情報提示部を1台のタブレットPCが兼ねる環境を構築した。

第4章 運動技能のセンシングデータを基に指導参考情報を提示する 教師用システムの開発と評価

学習者用端末部は17セット（スマートフォン34台，専用かんな17台）を使用した。モデルとなる熟練者のかんな掛けのデータは，前章で使用したもの（「3.4.1. 方法」に記載）と同一のものを使用した。

授業では，生徒は学習者用端末部を用いて各自で練習し（図4-9），教師は指導参考情報提示部を用いて練習履歴を確認しながら机間指導を行った（図4-10）。また，授業後に，学習者を対象に学習者用端末部の機能を評価する質問紙調査を，教師を対象に指導参考情報提示部を使用して，机間指導を行ったことに対する半構造化インタビューを行った。

なお，前章と同様に研究の限界として，本実験においては中学生を対象としており，教育の平等性の観点から統制条件を設定することは困難であった。



図4-9 学習者用端末部を用いて練習する生徒



図 4-10 指導参考情報提示部を用いて机間指導する教師

実験の手順について、授業前に前述の環境を整え、生徒の机上に削る木材および木材を固定するための台を配付した。授業の大まかな流れは表 4-1 の通りである。授業は 1 単位時間（50 分）で行われた。はじめに教師が既習事項の確認を行い、宮大工によるかんな掛けの動画（NHK『アインシュタインの眼 匠の技をとらえた！』）を視聴させたあと、デジタル教科書（間田ほか 2011）の動画教材（p.61『平けずりのしかた』）を用いて、生徒に熟練者のかんな掛けを確認させた。次に、学習者用端末部の操作方法を説明する動画を視聴させ、教師の演示により、学習者用端末部の操作手法とかんな掛けを指導した。その後、生徒が学習者用端末部を用いて各自でかんな掛けの練習に取り組んだ。このとき、生徒は 2 人で 1 グループを組んで練習を行った。教師から、5 回練習を行ったらペアの生徒と交代し、終了の合図があるまでそれを繰り返すよう指示した。最後に授業の感想等をまとめ、授業を終了した。授業終了後、生徒から質問紙を回収し（付録 1）、教師にインタビューを行った。

表 4-1 授業の流れ

導入 (5分)	<ul style="list-style-type: none"> ● 前時までの学習内容の確認させる ● 授業の目標を提示する
展開 (40分)	<ul style="list-style-type: none"> ● デジタル教科書の動画教材で熟練者のかんな掛けを見せる ● 学習者用端末部の操作方法を説明する動画を見せる ● 教師の演示により学習者用端末部の操作方法を確認させ、 かんな掛けを演示する ● 指導参考情報提示部を使用して机間指導をしながら、 学習者用端末部を用いて練習させる
まとめ (5分)	<ul style="list-style-type: none"> ● 授業の感想をワークシートに記入・発表させる ● 次時の予告をする

授業後に生徒から回収した質問紙の設問を表 4-2 に示す。この設問は、星野ほか(2003) および安藤・住川 (2013) による質問紙を参考に、本研究での実践に沿うように改変したものである。星野ほか (2003) は、「指導用教具には、『知識・理解』『興味・関心』『態度』『深化』『ストレス』『授業への集中度』という 6 つの機能があること」を明らかにした。また、安藤・住川 (2012) のように、星野ほか (2013) の知見を基に作成した質問紙で、開発した教材を技術・家庭科技術分野の授業において評価した例も見られる。そこで、本研究でも同様に、学習者用端末部の教材としての機能を「A.興味・関心」、「B.知識・理解」、「C.態度」、「D.深化」、「E.ストレス」、「F.集中度」の 6 つの観点から分析することとした。回答は「1 (いいえ) ～5 (はい)」の 5 件法で得た。「E.ストレス」および「F.集中度」の設問は全て逆転項目である。また、これらの項目に加え、授業の感想を自由記述で回答を得た。

表 4-2 質問紙の設問

観点	観点の概要	設問文
A.興味・関心	授業や学習内容への興味・関心の程度を評価する	1. かんなを使って木材を削ることは楽しいと思いましたが？ 2. もっと長く練習していたいと思いましたが？ 3. 意欲的に取り組むことができましたか？
B.知識・理解	学習内容の理解度を評価する	1. 正しいかんなの使い方を理解できましたか？ 2. 正しいからだ全身の使い方を理解できましたか？ 3. よく考えて作業できるようになりましたか？
C.態度	授業への取り組みの態度を評価する	1. 落ち着いた態度で作業をすることができましたか？ 2. 先生の説明をしっかりと聞けたと思いますか？ 3. 友達と余分な話をしないで授業を受けましたか？
D.深化	学習内容を深めることができたかを評価する	1. かんな掛けについて、もっと色々なことを知りたかったですか？ 2. 今日の授業で技能が身に付いたと思いますか？ 3. 今日のかんな掛けの練習を作品の製作に活かしていけると思いますか？
E.ストレス	授業や教材から受けるストレスの程度を評価する	1. 操作や使い方がわからなくなることがありましたか？ 2. 練習するときに嫌になることがありましたか？ 3. 練習するときに困ることがありましたか？
F.集中度	授業に熱心に取り組めたかを評価する	1. 授業の中で何をやるかわからなくなることがありましたか？ 2. 授業中に「ボーッ」とすることがありましたか？ 3. 授業にもっと集中できると良かったなと思いますか？

授業後に教師に行ったインタビューは、半構造化インタビューであった。教師用システムが教師の机間指導のしやすさに対してどのような効果があったかを調べることを目的とした。あらかじめ定めた質問は以下の4点であった。

1. 本システムを用いて授業を行った感想
2. 学習者用端末部が学習者のかんな掛けを評価した結果と、教師が目視で確認した結果に差はあったか
3. 本システムのどのような点を改善すべきか
4. 本システムがどのようなものに応用できそうか

4.3.2. 結果

生徒のかんな掛けの上達について、前章の評価方法と同様に、各生徒の最初の1回を開始時、最後の1回を終了時とし、開始時と終了時における3観点それぞれの得点の比較を行った。生徒の平均練習回数は9.8回であった。

使用した統計パッケージやそこで用いた関数、および固定要因やランダム要因の設定は前章と同様であった。（「3.4.2.結果」に記載）。

【①引きの速度】

分析の結果（図 4-11、表 4-3）、計測時点の主効果が有意（ $p<0.001$ ）であり、開始時（ $M=54.3$, $SD=28.2$ ）に比べ終了時（ $M=71.3$, $SD=26.4$ ）が高いことが確認された。練習回数の影響は見られなかった。

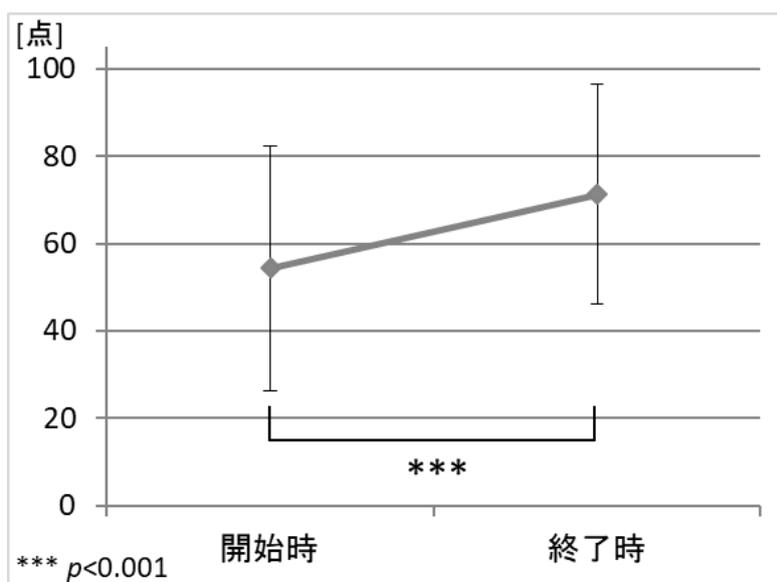


図 4-11 「①引きの速度」の結果

表 4-3 「①引きの速度」の結果

	推定値	標準誤差	t 値	p	
(切片)	61.58	7.13	8.64	<0.001	***
計測時点	17.00	2.59	6.58	<0.001	***
練習回数	0.13	0.69	0.18	0.855	

*** $p<0.001$

【②引きの直進性】

分析の結果（図 4-12，表 4-4），計測時点の主効果が有意（ $p=0.031$ ）であり，開始時（ $M=68.2$ ， $SD=28.1$ ）に比べ終了時（ $M=75.7$ ， $SD=26.0$ ）が高いことが確認された．練習回数の影響は見られなかった．

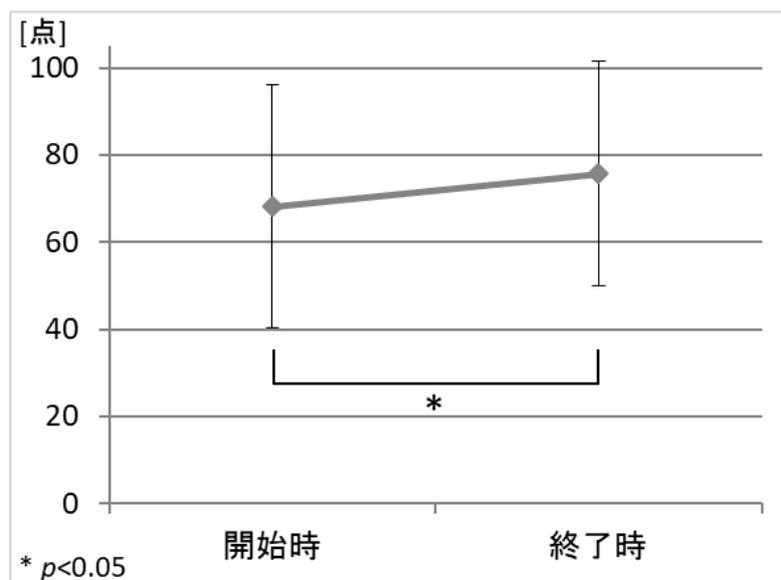


図 4-12 「②引きの直進性」の結果

表 4-4 「②引きの直進性」の結果

	推定値	標準誤差	t 値	p	
(切片)	77.71	6.28	12.37	<0.001	***
計測時点	7.55	3.46	2.19	0.031	*
練習回数	-0.59	0.61	-0.97	0.333	

*** $p<0.001$, * $p<0.05$

【③手元と腰の動き出しのタイミング】

第3章での分析と同様に、理想的な動作である「A.腰が先に動いてから手元を引いている」を「1」、不完全な動作である「B.腰も動いているが手元から引いている」および「C.手元だけで引いている」を「0」とし、ロジット混合モデルによって分析した。その結果（図4-13、表4-5）、練習開始時／終了時の主効果が有意（ $p=0.003$ ）であり、開始時に比べ終了時が高かいことが確認された。練習回数の影響は見られなかった。

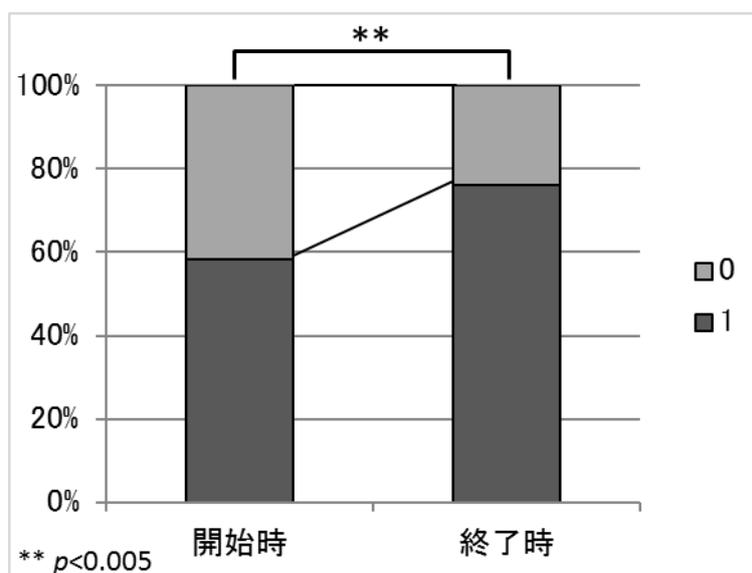


図 4-13 「③手元と腰の動き出しのタイミング」の結果

表 4-5 「③手元と腰の動き出しのタイミング」の結果

	推定値	標準誤差	z 値	p	
(切片)	0.05	0.82	0.06	0.952	
計測時点	1.27	0.43	2.96	0.003	**
練習回数	0.12	0.08	1.39	0.164	

** $p<0.005$

学習者用端末部の機能分析について、生徒から回収した質問紙の、各設問の平均値を図4-14に示す。「E.ストレス」および「F.集中度」の設問は逆転項目であったため、1が5、5が1となるように逆転処理を加えた。5件法によって回答を得たため、最低で1、最高で5となる。「A.興味・関心」においては、3つの設問全てにおいて、平均4.5以上と高い結果であった。一方、「E.ストレス」の「2.操作や使い方がわからなくなることがありましたか？」や、「F.集中度」の「3.授業にもっと集中できると良かったなと思いますか？」においては中間値の3は超えているものの、他の設問と比べると高いとはいえない結果であった。

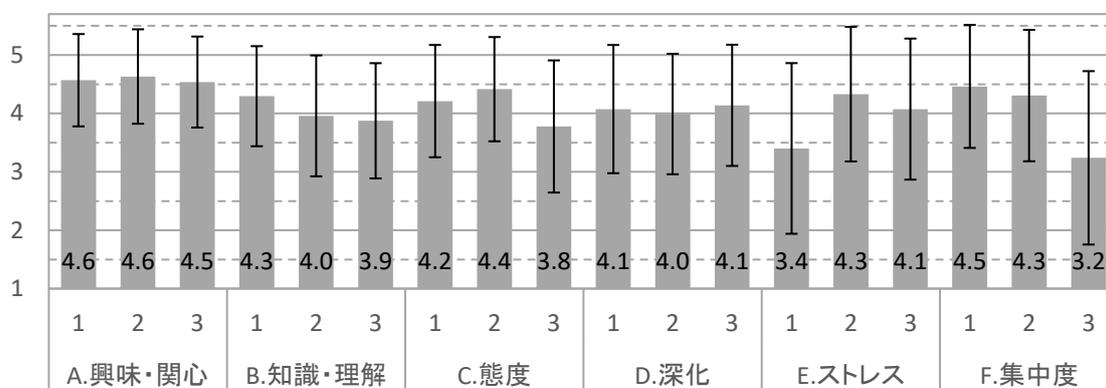


図 4-14 質問紙調査の各設問の平均値

教師へのインタビューによる本システムの評価について、「1.本システムを用いて授業を行った感想」では、学習者用端末部を使うことが、生徒が意欲的に取り組むきっかけになっていた点を第一に述べていた。また、指導参考情報提示部があったことで、かんな掛けが未熟な生徒をいち早く発見することができ、効率的な机間指導をすることができたとの回答を得た。具体的には、教師から遠い位置にいて目視では確認しづらい生徒の動作の状態を確認できた点が挙げられた。また、二人一組のペアで順番待ちをしている生徒の動作の状態は、本来その生徒に順番が回るまで確認することができないが、指導参考情報提示部があったことで、履歴データから確認することができた点も理由として挙げられた。

「2.学習者用端末部が学習者のかんな掛けを評価した結果と、教師が目視で確認した結果に差はあったか」の質問に対しては、両者の評価には特に差がなかったという回答を得た。

「3.本システムのどのような点を改善すべきか」の質問では、学習者用端末部に対して「遊んでしまう生徒がいる点」、「得点だけを見て、言葉によるアドバイスを読まない生徒がいる点」が挙げられ、「授業のやり方も含めて改善が必要だ」との回答を得た。

「4.本システムがどのようなものに応用できそうか」の質問に対しては、技術・家庭科技術分野でしばしば用いられる、のこぎりや玄翁などの工具に加え、体育の授業やラケット競技などが挙げられた。

4.3.3. 考察

生徒のかんな掛けは、3 観点全てにおいて有意な上達が確認された。前章で述べた学習者用アプリケーション単体での評価では見られなかった有意差も確認された。これについては、教師用システムを用いたことにより教師による机間指導が効率化されたことによるものと考察できる。後述するインタビュー調査からも、教師用システムの効果が示唆される結果が得られている。

学習者用端末部の機能分析について、「A.興味・関心」の結果が高かったことについては、自由記述の中にも「楽しかった」や「もっとやりたい」といった回答が多かったことから、生徒が学習者用端末部を用いて練習をすることで意欲的に取り組むことができたといえる。「F-3.授業にもっと集中できると良かったなと思いますか？」の結果が低かったことについては、速度の得点だけにこだわって、他の観点を無視して力任せにかんなを引いたり、学習者用アプリケーションの結果表示画面の「1. 引きの速度および加速度のグラフ」の波形が大きくなることを楽しんだりするなど、授業の本質ではない部分に熱中してしまっている生徒が一部見られた。後述する教師へのインタビューからも、「遊んでしまう生徒がいる」という同様の回答が得られた。「E-1.操作や使い方がわからなくなることがありましたか？」の結果が低かったことについては、学習者用アプリケーション起動時に行う2台のスマートフォンを Bluetooth で接続する操作や、ペアの生徒と交代するときに行う使用者を変更する操作が要因の1つではないかと考えられる。しかし、このことの裏付けとなる情報を、質問紙調査や教師へのインタビューなどから得ることができなかつたため、断定することはできない。生徒に対して、学習者用端末部の使用感についての調査を行うなどして、要因を明らかにする必要がある。

教師へのインタビューによる本システムの評価について、「1.本システムを用いて授業を行った感想」の回答からは、本システムが教師の机間指導の効率化に寄与することが示唆

されたといえる。また「2.学習者用端末部が学習者のかんな掛けを評価した結果と、教師が目視で確認した結果に差はあったか」の回答からは、本システムが生徒のかんな掛けを適切に評価することができていたことがうかがえる。「1.本システムを用いて授業を行った感想」および「3.本システムのどのような点を改善すべきか」の回答として得られた、「生徒が意欲的に取り組む点」や「遊んでしまう生徒がいる点」は、生徒を対象とした質問紙調査の結果と一致していた。遊んでしまう生徒がいる点については、学習者用端末部や授業の方法を改善することが課題である。

4.4. まとめと課題

本章では、学習者用アプリケーションを拡張し、取得したセンシングデータを基に指導参考情報を提示する教師用システムを開発し、授業実践を通して、運動技能の学習における連合段階および自動化段階の、対面による学習における学習支援が可能か評価を行った。その結果、取得したセンシングデータを基に指導参考情報を教師に提示することの価値が示唆され、生徒のかんな掛けも3観点全てにおいて有意な上達が確認された。従来、動画教材等を用いた「認知段階」の学習支援は行われていたが、運動技能のセンシングデータを活用することで連合段階および自動化段階の学習支援環境も実現できたといえる。対面による学習における支援という視点からは、従来、CAIで行われたような、個々の学習者の学習状況を教師が確認できる仕組みを、運動技能の特に連合段階および自動化段階の学習に対しても実現できたといえる。

開発した教師用システムでは、「1.グラフ表示」の中の「速度および加速度のグラフ」こそかんな掛けに特定した機能であるが、その他の、レーダーチャートの表示や「2.リスト表示」、「3.上達履歴表示」などの機能は、各観点の得点をそれぞれの表示方法で提示しているだけであるため、対象とする運動技能が変わり、評価する観点やその数が変わっても応用可能である。

教師用システムの評価は、インタビュー調査で得られた教師からの感想のみに基づいた考察に留まっている。指導参考情報がどのように活用され、何人の生徒にどのような個別指導が与えられたかなどの詳細な分析を行うことにより、システムの改良に有用な知見が得られると考えられるため、今後の課題としたい。

| 第4章 運動技能のセンシングデータを基に指導参考情報を提示する 教師用システムの開発と評価 |

本章においては、教師がその場に存在する対面による学習を対象としたため、取得したセンシングデータを、指導参考情報として教師へ提示することに活用したが、教師がその場に不在である e-Learning による学習を対象とする場合には、これとは異なる方法でセンシングデータを活用することが検討されるべきである。これについては第6章で述べる。

第5章 学習者用アプリケーションの模擬的な学習環境における評価

第3章で、専用かんや削るための木材、削り台などの実践的な学習環境が整った状態での、学習者用アプリケーションの効果を明らかにした。第4章における評価も同様に実践的な学習環境で行われた。本章では、ペットボトルをかんに見立ててシミュレーションによって学習する、模擬的な学習環境における効果について述べる。

5.1. 目的

前章までの評価は、専用かんや削るための木材、削り台など、必要なものが全て揃った実践的な学習環境におけるものであった。しかし、学習者にとって実践的な学習環境を整えることが困難であることも想定される。そのため、かんや木材などが揃っていない、模擬的な学習環境においても学習することが可能になれば、運動技能の学習における連合段階および自動化段階の学習環境を拡張することができる。

佐藤ほか(2008)は磁気式モーションキャプチャの弱点として「どこにでも移動できるものではなく、基本的には専用のスタジオで撮影が行われるなど制約が多い」と述べている。第3章で述べた学習者用アプリケーションの効果が模擬的な学習環境においても認められれば、この弱点が解消され、より簡便な手法であることが認められる。特に、e-Learningによる学習を想定した場合、e-Learningの利点の1つである「いつでもどこでも学習できる」ことは重要である。

本章では、学習者用アプリケーションを自宅での簡易的なシミュレーションによる練習に用いることによる、運動技能の学習における連合段階および自動化段階の学習支援に対する効果を評価する実験を行った。学習者にとっての、自宅に実践的な環境を整える負担を考慮し、一般家庭にある1.5リットルのペットボトルにスマートフォンを取り付けて、かん掛けのシミュレーションで、かん掛けの基礎的な感覚を身に付けるよう練習を行い、評価した。

5.2. 方法

実験参加者は大学生28名(男女各14名)であった。実験参加者の年齢は19~22歳で、身長は150cm~183cmであった。フィードバックが提示される学習者用アプリケーションを使用して練習を行うFBあり群と、フィードバックが提示されない学習者用アプリケ

ーションを使用して練習を行う FB なし群（各 14 名：男女各 7 名）の 2 つの実験群を設定した。

前章までの評価と同様に中学生を対象とすることを検討したが、本研究のような学習方法は前例がなく、学習への逆効果も懸念される。その場合の中学生への影響を考慮し、本実験の趣旨に理解を得られた大学生を対象とした。

中学生と大学生では体力や体格に差があるが、大学生や大人を対象とした先行研究としては、先行研究で用いられている作業台の高さに大差はなかった。田中・山崎（1990）が 680 mm、田中・安孫子（1990）が 700 mm、安孫子ほか（2005）が 710 mm の高さの作業台が用いられた。一方、中学生を対象とした先行研究としては、藤沖ほか（2013）が 670 mm の作業台が用いられており、ほぼ同程度の高さである。

また、近藤（1976）や橘田ほか（2003）のように、中学生と大人のかんな掛けを同条件で計測・比較している先行研究も見られる点から、大学生を対象とすることに問題はないと判断した。実施に当たっては、中学生、大学生問わず使うことができる寸法のかんなや、ひと息で削ることができる長さの被削材を用いた。また、かんな掛けの経験がない大学生を対象とすることで、中学生に近い条件で実験を行った。

本実験は、「1.事前調査」、「2.自宅での練習」、「3.事後調査」の 3 つの段階に分けて行った。

「1.事前調査」では、実験参加者の練習前のかんな掛けの状態を調査した。専用かんなを用いて動作をセンシングしながら 5 回のかんな掛けを行った。このとき、実験参加者へのフィードバックは提示されないようにした。

「2.自宅での練習」では、実験参加者がそれぞれの自宅で練習を行った。練習回数は 1 日 5 回、4 日間とした。このとき、FB あり群はフィードバックが提示される学習者用アプリケーションを使用し、それを参考にかんな掛けが上達するよう練習をした。FB なし群はフィードバックが提示されないものを使用し、中学校技術・家庭科技術分野の教科書を見ながら、記載されている情報を参考に、かんな掛けが上達するよう練習をした。

「3.事後調査」では、「1.事前調査」と同じ要領で、両群の自宅での練習後の状態を調査した。

「1.事前調査」と「3.事後調査」では、「①引きの速度」、「②引きの直進性」、「③手元と腰の動き出しのタイミング」の 3 観点における得点を記録した。本実験では、「1.事前調査」

と「3.事後調査」のそれぞれで記録した得点の比較と、実験後に行ったインタビュー調査から得られた回答を基に結果を考察することとした。

使用した機材等は、「1.事前調査」および「3.事後調査」において、専用かんな、スマートフォン、削るための棒材、棒材を固定する削り台であった。棒材は長さ 400 mm のものを用いた。使用した机の高さは 700 mm であった。「2.自宅での練習」においては、スマートフォン、および 1.5 リットルのペットボトルを用いた。図 5-1 にスマートフォンを取り付けた 1.5 リットルのペットボトルを示す。0.9 リットルや 1 リットルのペットボトルよりは身近であり、2 リットルのペットボトルよりは大きさや重さがかんなに近いという理由から選定した。ペットボトルは開封前の中身が入った状態のものを用いた。また作業台の高さや材料の長さについては、自宅での練習時にも「1.事前調査」および「3.事後調査」との環境を極力揃えられるよう、高さ 700 mm 程度の机で、400 mm の長さの材料を削ることを意識して練習するよう実験参加者に指示した。

モデルとなる熟練者のかんな掛けのデータは、「3.4.学習者用アプリケーションの評価」で使用したものと同一のものを使用した。



図 5-1 スマートフォンを取り付けたペットボトル

5.3. 結果

前章までの分析と同様の手法を用いて（「3.4.2.結果」に記載），3 観点それぞれの得点の分析を行った。

【①引きの速度】

実験群（FB あり群／FB なし群）と計測時点（開始時／終了時）を固定要因とした。ランダム要因は、実験参加者の切片のみを設定した。分析の結果（図 5-2，表 5-1），実験群×計測時点の交互作用が有意であった（ $p=0.027$ ）。交互作用に関して下位検定を行った結果，FB あり群における計測時点の単純主効果が有意（ $p<0.001$ ）であり，FB なし群における計測時点の単純主効果が有意傾向（ $p=0.084$ ）であった。また，練習後における実験群の単純主効果が有意傾向（ $p=0.067$ ）であった。練習前においては，実験群の単純主効果は有意ではなかった。

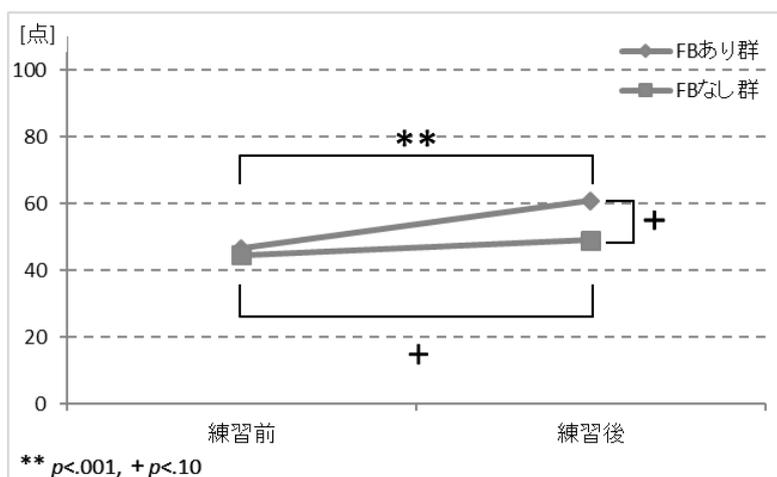


図 5-2 「①引きの速度」における得点の変化

表 5-1 「①引きの速度」における得点の変化

	推定値	標準誤差	t 値	p	
(切片)	44.28	4.30	10.29	<0.001	***
計測時点	5.10	2.94	1.74	0.084	+
実験群	2.34	6.06	0.39	0.702	
計測時点×実験群	9.14	4.11	2.23	0.027	*

*** $p<0.001$, * $p<0.05$, + $p<0.10$

【②引きの直進性】

①の分析と同様の固定要因およびランダム要因を設定したモデルを採用した。分析の結果（図 5-3、表 5-2）、計測時点および実験群のいずれの要因の単純主効果も有意ではなかった。

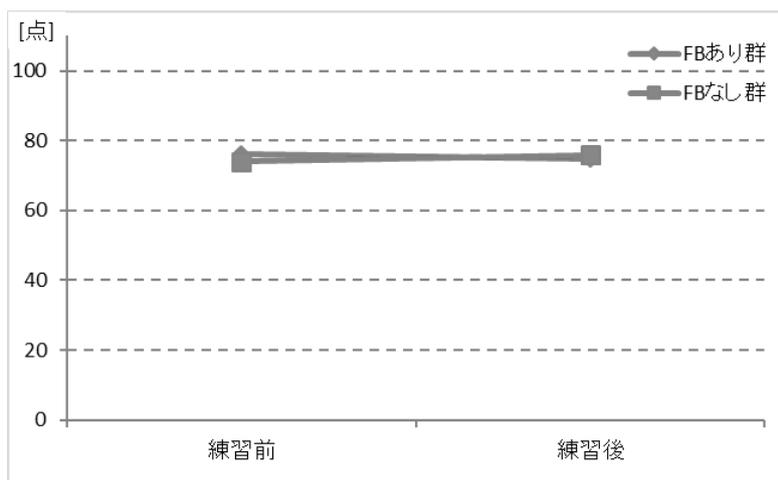


図 5-3 「②引きの直進性」における得点の変化

表 5-2 「②引きの直進性」における得点の変化

	推定値	標準誤差	t 値	p	
(切片)	73.53	4.13	17.81	<0.001	***
計測時点	2.42	4.11	0.59	0.556	
実験群	1.74	5.79	0.30	0.766	
計測時点×実験群	-2.86	5.75	-0.59	0.619	

*** $p < 0.001$

【③手元と腰の動き出しのタイミング】

第3章および第4章での分析と同様に、理想的な動作である「A.腰が先に動いてから手元を引いている」を「1」、不完全な動作である「B.腰も動いているが手元から引いている」および「C.手元だけで引いている」を「0」とした。また、①および②の分析と同様の固定要因およびランダム要因を設定したモデルを採用した。分析の結果(図5-4, 表5-3), 実験群×計測時点の交互作用が有意傾向であった ($p=0.069$)。交互作用に関して下位検定を行った結果, FBあり群における計測時点の単純主効果が有意 ($p=0.011$) であった。FBなし群における計測時点の単純主効果および練習前後における実験群の単純主効果はいずれも有意ではなかった。

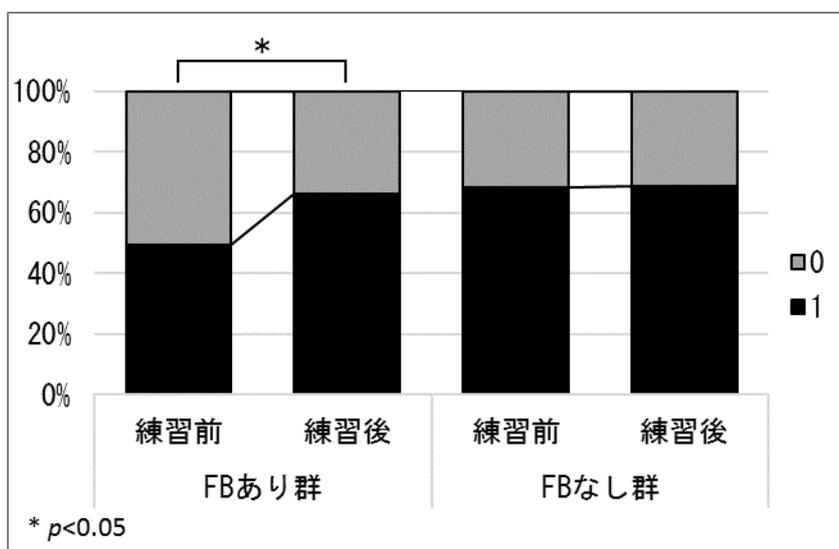


図 5-4 「③手元と腰の動き出しのタイミング」の結果

表 5-3 「③手元と腰の動き出しのタイミング」の結果

	推定値	標準誤差	z値	p	
(切片)	1.11	0.55	2.01	0.045	*
計測時点	0.02	0.43	0.05	0.959	
実験群	-1.24	0.77	-1.60	0.110	
計測時点×実験群	1.13	0.62	1.82	0.069	+

* $p<0.05$, + $p<0.10$

実験参加者からの意見としては、実験参加者自身の実感として、上達を感じたとの意見がFBあり群から多く挙げられた。一方、FBなし群からは、教科書に書いてある情報を意識しながら練習はしたものの、自分の動作が実際にどのようなになっているのかわからず、その状態で良いのか、教科書通りの動作になっているのかわからなかったという意見が多かった。また、FBあり群の方がFBなし群よりも楽しみながら練習でき、練習後の動作の上達や、練習することの意義を感じている傾向にあることがわかった。

5.4. 考察

本実験の結果からは、FBあり群において、「①引きの速度」および「③手元と腰の動き出しのタイミング」において有意な上達が確認された。「②引きの直進性」においては、有意な得点の変化は見られなかった。また、FBなし群において、「①引きの速度」の上達に有意傾向が確認された。

「①引きの速度」および「③手元と腰の動き出しのタイミング」については、田中ほか(1990)が「熟練者は、かんな引き始めまでに腰を素早く引いて、かんなの動き出しを助けている」と述べているように、全身を使って腰から引くことが引きの速度の影響を与えていると考えられるため、両観点FBあり群において共に有意に上達したと考察できる。

「③手元と腰の動き出しのタイミング」において練習後の群間における有意な差は確認されなかった点について、各実験参加者を個別に分析した。本実験では練習前後にそれぞれ5回ずつかんな掛けのセンシングをしたが、その中の「A.腰が先に動いてから手元を引いている」動作ができていた回数に着目した。その結果、FBあり群において、「A.腰が先に動いてから手元を引いている」動作ができていた回数が増加していた者が11名、変化のない者が1名、減少していた者が2名であったのに対し、FBなし群では、増加していた者が8名、変化のない者が0名、減少していた者が6名であった。このことから、練習後における群間の有意な差は認められなかったものの、「③手元と腰の動き出しのタイミング」におけるフィードバックにある程度の効果があったことが想定される。

また、「②引きの直進性」の得点には有意な変化は見られなかったが、この点については次のような考察が考えられる。かんなを単に速く引こうとした場合、力を込めて思い切り引くことになるため、引きの直進性が低下することが予想される。しかし、FBあり群において「①引きの速度」の得点が有意に向上している一方で、「②引きの直進性」の得点が低下しなかった。このことを考慮すると、さらなるフィードバックの方法の改善が求めら

れるものの、「②引きの直進性」におけるフィードバックにもある程度の効果があったことが想定される。

FB なし群において、「①引きの速度」の上達に有意傾向が確認されたことについては、次のように考察することができる。FB なし群は、学習者用アプリケーションからのフィードバック（外在的フィードバック）こそ受けていないが、「1.事前調査」と「3.事後調査」で計 10 回、専用かんなを使って削っており、切削抵抗に負けずに引ききれたかどうか、削りくずがどの程度出てきたか、という情報（内在的フィードバック）を得ることができる。これを受けて自身のかんな掛けを修正しようとした結果であると考えられる。

しかし、「①引きの速度」における練習後の群間の差が有意であったことや、「③手元と腰の動き出しのタイミング」において FB あり群のみ有意な上達が確認されたことから、学習者用アプリケーションによるセンシング・フィードバックが有効であったことが認められる結果であったといえる。

5.5. まとめと課題

学習者用アプリケーションの模擬的な学習環境における評価として、1.5 リットルのペットボトルをかんなに見立てたシミュレーションによる練習を行い、フィードバックの有無および練習前後のかんな掛けを比較する実験を行った。その結果、評価している 3 観点のうち、「①引きの速度」および「③手元と腰の動き出しのタイミング」の 2 観点において上達が確認された。身近なデバイスであるスマートフォンを用いたセンシングや、取得したデータに基づいたフィードバックの提示は、模擬的な学習環境においても、運動技能の学習における連合段階および自動化段階の学習支援に対して一定の効果があることが示された。

この知見は、専用の環境が整った場でなくても運動技能の学習が可能であるという点で意義があるといえる。また、「いつでもどこでも学習できる」という利点をもつ e-Learning による学習との相性も良く、組み合わせることにより更なる学習支援を行うことができると考えられる。具体的には、汎用の LMS と連携して学習管理ができるようになれば、認知段階の学習と組み合わせたり、運動技能の習得の程度に合わせて適切な教材を提示したりすることが可能になる。これについて、第 6 章で述べる。

また、本章における評価では2台のスマートフォンを用いたが、模擬的な学習環境としてさらに簡便な装備での学習効果を評価しようとした場合、かんな端末と腰端末のどちらか1台のみを用いた評価も考えられる。装備の簡便化のために評価する観点を絞る場合、どの観点が全観点の結果や削られた木材の仕上がりに強く影響しているかを明らかにすることにより、より妥当な絞り込みができると考えられる。

他の運動技能へ応用する視点では、本研究ではかんなに近いと考えられるペットボトルを用いたが、他の運動技能でも、例えば傘をゴルフクラブに見立ててスイングの練習をすることなどが考えられる。これらは今後の課題としたい。

第6章 運動技能の学習を汎用のLMSで管理するためのプラグインの開発

前章では、学習者用アプリケーションが模擬的な学習環境においても有用であることを述べた。本章では、e-Learningによる学習における学習支援を検討することとした。e-Learningによる学習では、LMSにより学習が管理されることにより、教材単体を用いた学習以上の利点を得ることができる。そこで、学習者用アプリケーションとLMSを連携させるプラグインの開発を行った。LMSにはMoodleを用いた。

なお、本章では、学習者用アプリケーションおよびプラグインのソースコードのうち、主要な箇所を抜き出して解説を加えている。第3章での学習者用アプリケーションの開発、および第4章での教師用システムの開発においては、開発したシステム等による学習支援への効果の評価に主眼を置いたため、具体的なソースコードは付録にのみ記載した。一方、本章でのプラグインの開発においては、Moodleと学習者用アプリケーションの連携やデータの受け渡しをどのように実現したのかが重要であると考え、付録へのソースコードの記載に加えて、本文中でもソースコードの解説を記載することとした。

6.1. 目的

開発した学習者用アプリケーションは、実践的な学習環境においても、模擬的な学習環境においても有用であることが示された。この知見は、換言すると、スマートフォンという身近なデバイスを用いて取得したセンシングデータを基にフィードバックを提示する支援の方法により、実践的な学習環境の有無にかかわらず、学習者が学習したいときに、学習したい場所で学習できることの可能性が示唆されたといえる。これは、「いつでもどこでも学習できる」というe-Learningによる学習の利点(高岡 2012)とも馴染むものである。第2章で述べたように、e-Learningによる学習をLMSで管理することには、「演習問題に誤った場合、正解と解説を提示するだけでなく、該当コンテンツへジャンプできる機能(リンク)などが有用である」(植野 2007)などの利点がある。学習者用アプリケーションを用いた運動技能の学習も、LMSと連携して学習を管理することができれば、e-Learningによる学習も支援することができると考えられる。外部のシステムとの連携のような、LMSの機能を拡張させる方法として、プラグインの開発が挙げられる(赤倉・柏原 2016)。本章では、学習者用アプリケーションにより取得したセンシングデータを基に、汎用のLMSで学習を管理するためのプラグインを開発することを目的とした。

6.2. プラグインの開発

6.2.1. プラグインの設計

汎用のLMSとして、世界で最も多く利用されており、またオープンソースで利用や拡張が容易なMoodleを用いることとした(Moodle Pty Ltd. 2017)。

Moodleには、「小テスト」と呼ばれる、1つまたは複数の問題から構成されるモジュールが存在する。「小テスト」では、様々な「問題タイプ」のプラグインが使用される。標準機能としても多様な「問題タイプ」のプラグインが存在する。例えば、「多肢選択問題」や「○/×問題」、「記述問題」などが挙げられる。これらを用いることにより、いずれも出された問題に対して学習者が解答することができ、学習内容の習得の状況を記録・管理することができる。また、その習得の状況に応じて、その後の学習に適切な教材を提示することなどが可能である。

運動技能の学習に対して「小テスト」の機能を用いることを、運動技能の学習における認知段階、連合段階、自動化段階の視点から整理する。認知段階は何を行うかを理解して言語的に戦略を考える段階であるため(嘉戸・伊藤 2008)、従来の「情報」や「心的手続き」が中心であったe-Learning(板垣・堀田 2016)で用いられていた既存の問題タイプが活用可能である。一方、連合段階および自動化段階においては、渡辺(2015)が「筋肉を使って体を動かす/コントロールする力」の評価方法として「身体活動を実際に行って、結果を測定し評価する」と挙げていることを参考にすれば、試行して計測した結果、すなわち学習者用アプリケーションが算出した得点と「小テスト」の機能が連動することにより実現可能であると考えられる。

開発するプラグインは、第3章で述べたかんな掛けを対象とした学習者用アプリケーションだけでなく、様々な運動技能に対して応用可能であることが望ましい。それぞれの運動技能によって、評価する観点やその数は異なると考えられる。評価する観の違いについては、それぞれの運動技能に対応した学習者用アプリケーションが必要となるが、その数については、LMS側で柔軟に対応できることが望ましい。

以上を踏まえると、運動技能の学習における連合段階および自動化段階の学習をLMSで管理するための要件は、以下の3点に整理できる。

- 要件1. 対象とする運動技能におけるいくつかの然るべき観点において運動技能をセンシングし、各観点の得点を算出する学習者用アプリケーションがあること
- 要件2. LMS上で評価する観点の数に応じた問題を作成できること
- 要件3. 要件1で算出された得点が、要件2で作成された各問題に対応した解答となるように連携すること

本章では、要件1には、第3章で述べたかな掛けを対象とした学習者用アプリケーションを用いることとした。そして、要件2および要件3を満たすプラグインを開発することとした。

開発にあたっては、Moodleの標準機能として備わっている問題タイプのプラグインである「数値問題」に基づいて、その一部を改変して開発することとした。標準機能として備わっているプラグインに基づいたのは、既存の機能と使用感が近い方が利用しやすいと考えたためである。数値問題のプラグインに基づいたのは、学習者用アプリケーションが算出する得点が数値であり、数値問題のプラグインと相性が良いと考えたためである。

6.2.2. プラグインの概要

使用したMoodleのバージョンは、プラグインの開発時(2017年6月)に最新版であったMoodle 3.3であった。レンタルサーバ上にインストールして、Moodleを使用できる環境を構築した。レンタルサーバのOSはFreeBSD、WebサーバソフトウェアはApache、データベースサーバソフトウェアはMySQLであった。

本プラグインのMoodleへのインストールは、Moodleのサーバ内の然るべきディレクトリ(通常は、「<Moodleのインストールディレクトリ>/question/type」)に必要なファイルをディレクトリごと配置し、管理者アカウントでログインした上でMoodleの「サイト管理」のページにアクセスし、インストールを実行することにより行うことができる。この手順は一般的なプラグインのインストール時と同様である。

開発したプラグインの機能の概要を図6-1に示す。本プラグインにより、教師による運動技能の問題の作成を行うことができ、また、学習者用アプリケーションにより算出された得点をMoodleで管理するための橋渡しが行われる。

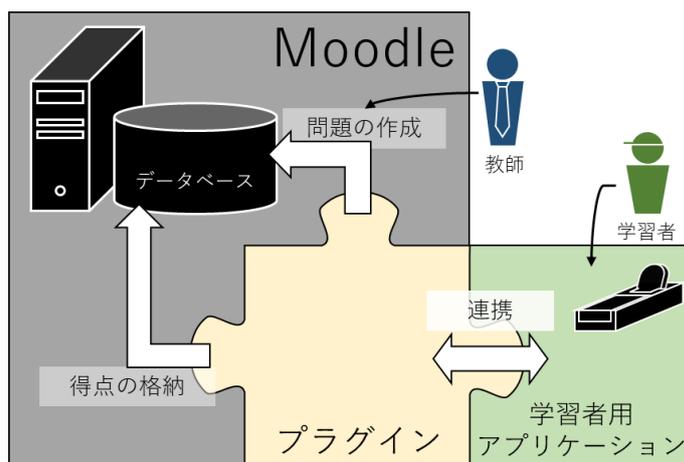


図 6-1 開発したプラグインの機能の概要

開発したプラグインの使用時の流れを図 6-2 に示した。教師は管理者としてログインすることで、プラグインのインストールや、運動技能の問題の作成を行うことができる。学習者は、小テストのページへアクセスし、受験することができる。本文では具体的に、「教師による運動技能の問題の作成」、「学習者による受験時のプラグインの機能」、「学習者による受験時のデータの流れ」の3つに分けて述べる。

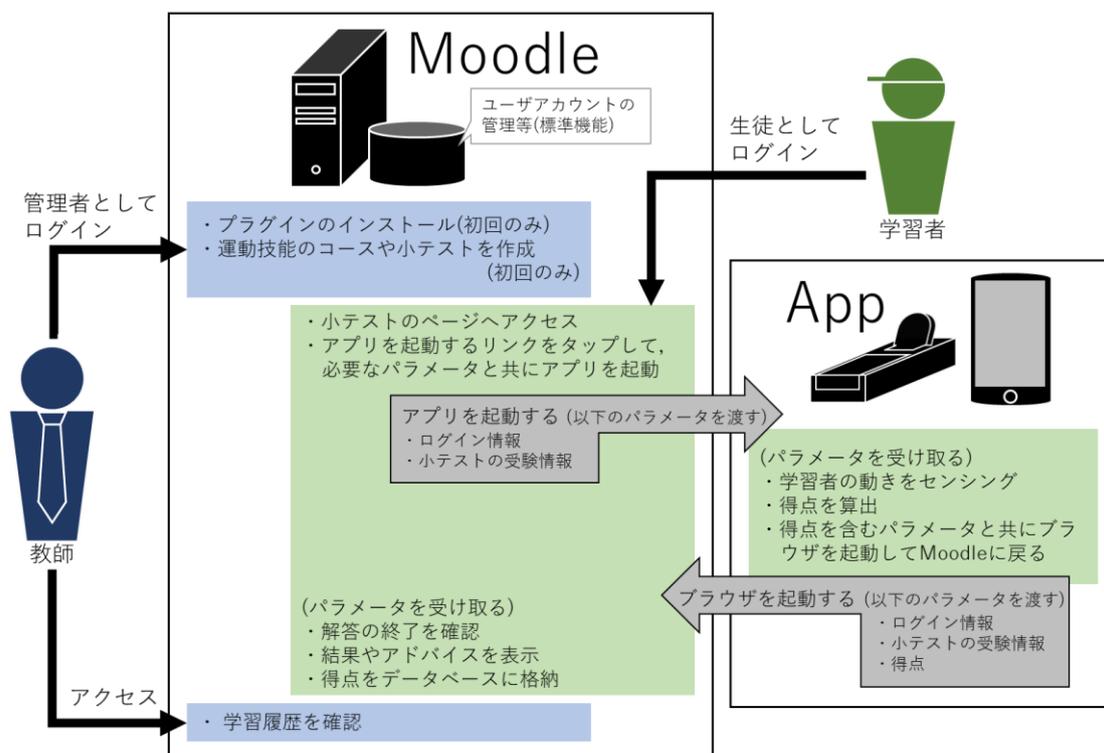


図 6-2 開発したプラグインの使用時の流れ

教師による運動技能の問題の作成について、教師は評価する観点の数だけ問題を作成する。例えばかんな掛けの場合、評価する観点の数は3つであるため、3つの問題を作成する。問題作成時の流れを図6-3に示す。

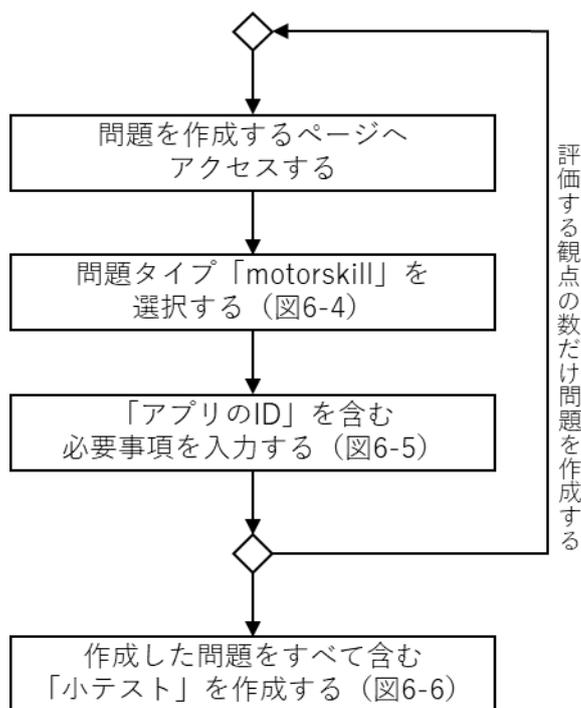


図 6-3 問題作成時の流れ

はじめに新しい問題を作成するページで、「motorskill」の問題タイプを選択する（図6-4）。その次に、「問題名」、「問題テキスト」、「アプリのID」、「答え」、「誤差」、「評点」、「フィードバック」を入力する（図6-5）。「アプリのID」以外は、標準機能として備わっている「数値問題」の機能である。「アプリのID」は、起動する学習者用アプリケーションを指定するものである。詳細は後述する。「答え」、「誤差」、「評点」、「フィードバック」はそれぞれ複数指定することができる。例えば、100点満点の「引きの速度」において、100点～80点の場合は評点が3で「その調子です」というフィードバックを、80点～60点の場合は評点が2で「もう少し早く引きましょう」というフィードバックを、60点未満の場合は評点が1で「まだまだ引きの速度が足りません」というフィードバックを提示する、といった指定をすることができる。

評価する観点の数だけ問題を作成したら、それらを含む小テストを作成する（図6-6）。



図 6-4 問題作成時に問題タイプを選択するページ

Adding a motorskill question ⓘ

▶ すべてを展開する

▼ 一般

現在のカテゴリ
現在のカテゴリ 木材加工の技能 のデフォルト (9) このカテゴリを使用する

カテゴリに保存する
木材加工の技能 のデフォルト (9) ▼

問題名 ⓘ
[かんな3]手元と腰の動き出しのタイミング

問題テキスト ⓘ
【アプリで練習】 [かんな]手元と腰の動き出しのタイミング

アプリのID ⓘ
kanna

▼ 答え

Answer 1
Answer 1 3 Error 0 評点 100% ⇅

フィードバック
完璧です！腰から全身で引けています。

図 6-5 問題作成時に諸項目を入力するページ



図 6-6 各観点の問題で構成された小テスト

学習者による受験時のプラグインの機能について、受験時の流れの視点から、標準機能として備わっている「数値問題」のプラグインの機能と比較したものを図 6-7 に示す。網掛け部分が「数値問題」から修正した部分である。また、学習者がアクセスする小テストのページを図 6-8 に示す。

標準機能として備わっている問題タイプのプラグインでは、ページ上のフォームに解答を入力し、送信すると解答に応じたフィードバックが提示される。それに対し、開発したプラグインでは、フォームへの解答の入力を学習者用アプリケーションがフックする。そのために、ページ上のフォームを無効化して入力できないようにし、学習者用アプリケーションを起動するためのリンクを表示するよう開発した。このリンクをタップすると、Moodle から一度離れ、学習者用アプリケーションが起動する。学習者がアプリケーションを用いて練習を行うと、学習者用アプリケーションが終了し、Moodle へ戻る。戻った Moodle 上の小テストのページでは、学習者用アプリケーションが算出した各観点の得点が解答としてフォームに入力された状態になる(図 6-9)。この後の、解答の送信やフィードバックの提示、データベースへの得点の格納は標準機能として備わっているものと同じである。

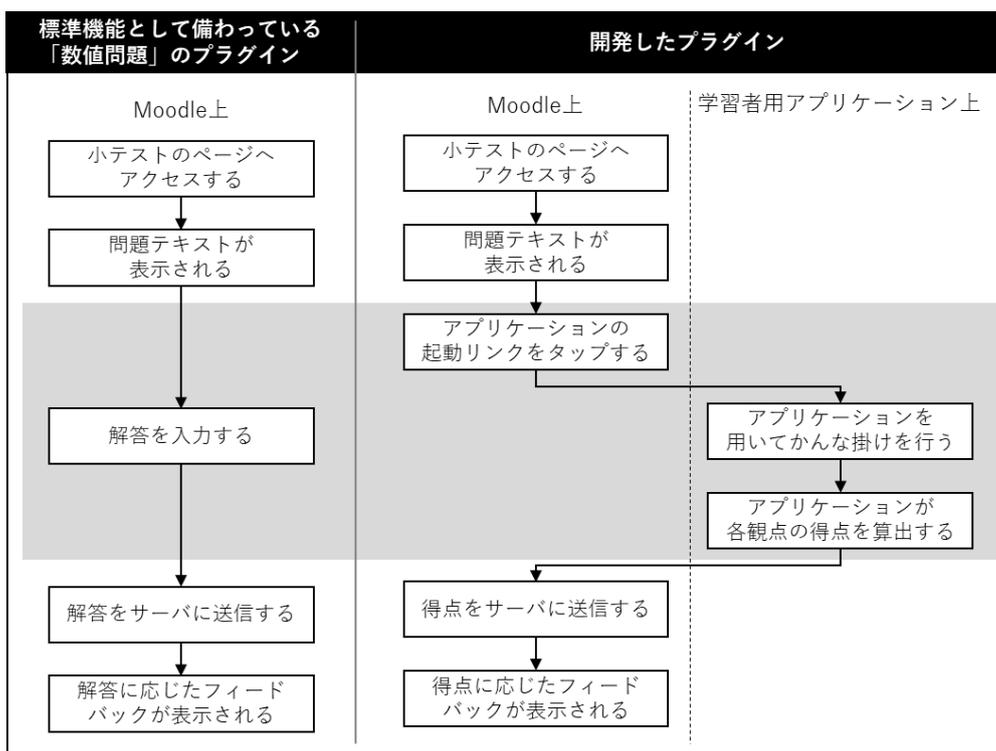


図 6-7 学習者による小テスト受験時の流れ

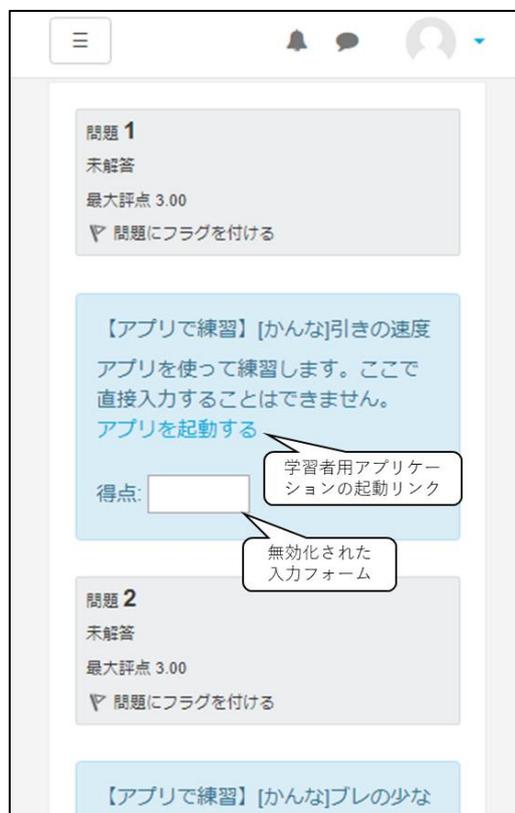


図 6-8 本プラグインを使用した小テストのページ

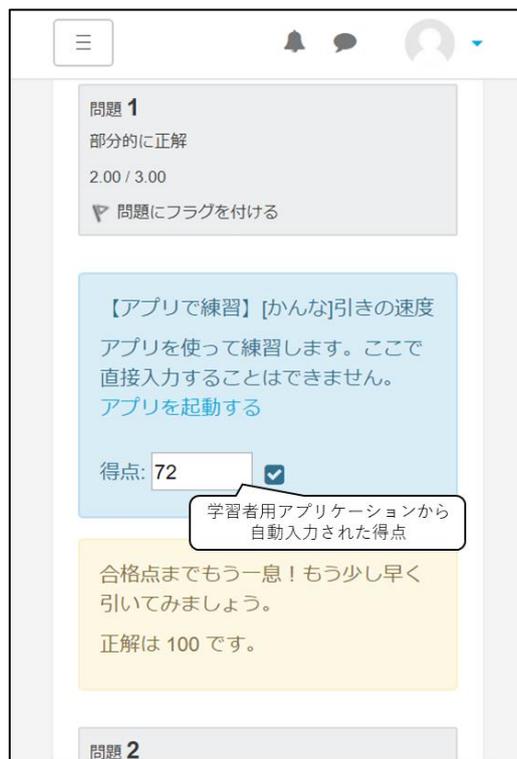


図 6-9 学習者用アプリケーションが算出した得点が入力された状態

学習者による受験時のデータの流れについて、当該箇所のソースコードに基づいて述べる。学習者用アプリケーションを起動するリンクのソースコードをコード 6-1 に示した。「scheme」とは、起動する学習者用アプリケーションを指定する ID である。このリンクをタップすると、同一の「scheme」が指定されているアプリケーションが起動される。アプリケーション側ではコード 6-2 のように指定する。コード 6-2 では、「scheme」に「kanna」を指定している。図 6-5 で示した問題作成時に諸項目を入力するページにおいて、教師が「アプリの ID」として入力するのが、この「scheme」であり、同様に「kanna」と入力することでこのアプリケーションが起動する。

```
01. <a href='intent://whatever/#Intent;scheme=".$app.";S.ssesskey=".$tempssesskey.";S.attempt=".$tempattempt.";S.usageid=".$usageid.";end'>アプリを起動する</a>
```

コード 6-1 学習者用アプリケーションを起動するリンク

```
01. <data android:scheme="kanna"/>
```

コード 6-2 学習者用アプリケーションでの「scheme」の指定

コード 6-1 に示したリンクをタップすると、ログイン情報である「sesskey」と、小テストの受験情報である「attempt」および「usageid」が学習者用アプリケーションに受け渡される。学習者用アプリケーション側でのこれらのデータの受け取り部分のソースコードはコード 6-3 である。2 行目の条件分岐では、Moodle の小テストのページから学習者用アプリケーションが起動されたかどうかを確認している。学習者用アプリケーションが不正に起動された場合は、9 行目の処理によって「このアプリは Moodle から起動して使用します。」というエラーメッセージが表示される。学習者用アプリケーションが Moodle の小テストのページから正常に起動された場合は、4 行目から 6 行目の処理によって、各パラメータが学習者用アプリケーション側で受け取られる。

```

01. // Moodle から起動されたことを確認
02. if (String.valueOf(getIntent().getCategories()).equals("{android.intent.category.
    BROWSABLE}")){
03. // 各パラメータの受け取り
04. sesskey = extras.getString("sesskey");
05. attempt = extras.getString("attempt");
06. usageid = extras.getString("usageid");
    ~略~
07. } else {
08. // エラーメッセージの表示
09. Toast.makeText(this, "このアプリは Moodle から起動して使用します。", Toast.LENGTH_
    LONG).show();
10. }

```

コード 6-3 学習者用アプリケーション起動時の Moodle からのデータの受け取り

学習者が練習を終えると学習者用アプリケーションが算出した各観点の得点である「answers」が、「sesskey」、「attempt」および「usageid」と共に Moodle へ受け渡される（コード 6-4）。2 行目では、受け渡し先として、Moodle 側の受け取り用のプログラムを指定している。4 行目では受け渡すパラメータを記述している。なお、このとき「answers」は、評価する各観点の得点をコンマ続きの形式で記述しておく、例えば、「引きの速度」が 90 点、「引きの直進性」が 80 点、「手元と腰の動き出しのタイミング」が 3 段階評価のうちの「3」だった場合、「answers = 90,80,3」となる。

```

01. // Moodle側の受け取り用のプログラムを指定
02. Uri uri = Uri.parse(moodlehost + "question/type/motorskill/posttomoodle.php?
    answers="+answers+"&sesskey="+sesskey+"&attempt="+attempt+"&usageid="+usageid);
03. // 受け渡すパラメータを記述
04. Intent intent = new Intent(Intent.ACTION_VIEW, uri);
05. byte[] postdata = ("answers="+answers+"&sesskey="+sesskey+"&attempt="+attempt).getBytes();
06. Intent intent2 = intent.putExtra("com.android.browser.post_data", postdata);
07. startActivity(intent2);

```

コード 6-4 学習者用アプリケーションが算出した得点の Moodle への受け渡し

Moodle 側での受け取り部分のソースコードはコード 6-5 である。コンマ続きの文字列として受け渡された「answers」は、受け取り側で配列データとして扱い（2行目）、ループで処理される（4行目から6行目）。ループの繰り返しの回数は、1行目で受け取られた配列データの要素数である。すなわち、評価する観点や作成された問題数と同一である。そのため、対象とする運動技能が変わり、評価する観点の数が変わっても、その運動技能の各観点の得点を算出する学習者用アプリケーションがあれば本プラグインを用いることができ、運動技能の学習を Moodle で管理することができる。

なお、ログイン情報である「sesskey」や、小テストの受験情報である「attempt」や「usageid」が共に受け渡されることにより、不正なアクセスを防ぐことができる。この処理については、Moodle に備わっている標準機能によるものであるため、本論文においては省略する。

```

01. // 受け取ったコンマ続きの得点を配列に変換
02. $answer_array = explode(",", $answers);
    ~略~
03. // 得点をループで処理
04. for ($i=0; $i<count($answer_array); $i++) {
05. $j = $i+1;
    ~略~
06. $html .= "<input type='hidden' name='q'.".$usageid.".'" . $j . "_answer' value='". $answer_array[$i] . "'>";
    ~略~
07. }

```

コード 6-5 学習者用アプリケーションが算出した得点の Moodle での受け取り

表 6-1 は、基にした「数値問題」のプラグインの、どのファイルに、どのような修正を加えたのかをまとめたものである。「◆」はディレクトリを、「◇」はファイルを表している。「置換のみ」は、コード内の「numerical」の部分をも「motorskill」に置換しただけの、実質的に修正のないファイルである。「コードの修正あり」は、この置換に加え、コードに加筆・修正を加えたファイルである。「新規作成」は、必要に応じて新たに作成したファイルである。「画像差替」は、アイコン用の画像を差し替えたことを示す。以下に「コードの修正あり」および「新規作成」のファイルを取り上げ、ソースコードを解説する。なお、ソースコードの全体については付録を参照されたい。

表 6-1 「numerical」からの修正

ディレクトリ (◆) およびファイル (◇)	修正
◆motorskill	—
├─◆backup	—
│ ├─◆moodle1	—
│ │ └─◇lib.php	置換のみ
│ └─◆moodle2	—
│ │ └─◇backup_qtype_motorskill_plugin.class.php	置換のみ
│ │ └─◇restore_qtype_motorskill_plugin.class.php	置換のみ
├─◆db	—
│ └─◇install.xml	コードの修正あり
│ └─◇upgrade.php	置換のみ
│ └─◇upgradelib.php	置換のみ
├─◆lang	—
│ └─◆en	—
│ │ └─◇qtype_motorskill.php	置換のみ
├─◆pix	—
│ └─◇icon.gif	画像差替
├─◆tests	—
│ └─◇answerprocessor_test.php	置換のみ
│ └─◇form_test.php	置換のみ
│ └─◇helper.php	置換のみ
│ └─◇question_test.php	置換のみ
│ └─◇questiontype_test.php	置換のみ
│ └─◇upgradelibnewqe_test.php	置換のみ
│ └─◇walkthrough_test.php	置換のみ
└─◇edit_motorskill_form.php	コードの修正あり
└─◇lib.php	置換のみ
└─◇question.php	置換のみ
└─◇questiontype.php	コードの修正あり
└─◇renderer.php	コードの修正あり
└─◇styles.css	置換のみ
└─◇version.php	置換のみ
└─◇posttomoodle.php	新規作成

「motorskill/db/install.xml」は、プラグインを新規にインストールする際に、必要なテーブルをデータベースに作成するための情報が書かれたファイルである。このファイルに、コード 6-6 に示したものを加筆した。コード 6-6 には、起動する学習者用アプリケーションを指定する ID である「scheme」を格納するテーブルを作成するための情報が書かれている。「scheme」を指定する機能は「数値問題」のプラグインには存在しなかったため、コード 6-6 を追記する必要がある。2 行目ではテーブル名を指定している。4 行目から 7 行目ではテーブル内に作成するカラムを指定している。9 行目から 11 行目では、他のテーブルと連携する際に必要な主キーを指定している。

```

01. <!-- テーブル名を指定 -->
02. <TABLE NAME="question_motorskill_app" COMMENT="App ID for practice of motorskill">
03. <!-- テーブル内に作成するカラムを指定 -->
04. <FIELDS>
05. <FIELD NAME="question" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0"
    SEQUENCE="false"/>
06. <FIELD NAME="app_id" TYPE="char" LENGTH="255" NOTNULL="true" DEFAULT="app"
    SEQUENCE="false"/>
07. </FIELDS>
08. <!-- 主キーを指定 -->
09. <KEYS>
10. <KEY NAME="primary" TYPE="primary" FIELDS="question"/>
11. </KEYS>
12. </TABLE>

```

コード 6-6 「motorskill/db/install.xml」に加筆したソースコード

「motorskill/edit_motorskill_form.php」は、運動技能の問題を作成したり編集したりするときに呼び出されるファイルである。このファイルに、コード 6-7 に示したものを加筆した。コード 6-7 の加筆により、問題作成時に、「scheme」を入力するフォームが表示されるようになる。「scheme」を指定する機能は「数値問題」のプラグインには存在しなかったため、コード 6-7 を追記する必要がある。ここで入力された「scheme」がコード 6-6 によって作成されたテーブルに格納される。

表示するフォームを生成しているのは 9 行目である。既存の問題を編集する際には、3 行目から 7 行目の処理で取得された、入力済みの「scheme」がフォーム内に表示される。問題を新規作成する場合は空欄となる。

```

01. protected function add_app_id($mform) {
02. // データベースに接続
03. global $DB;
04. $id = $_GET['id'];
05. // 既存の scheme があれば取得
06. $res = $DB->get_record("question_motorskill_app", array('question' => $id));
07. $app = $res->app_id;
08. // フォームの生成
09. $mform->addElement('text', 'appid', 'アプリの ID', array('size' => 50, 'maxlength'
=> 255, 'value' => $app));
10. $mform->setType('appid', PARAM_TEXT);
11. $mform->addRule('appid', null, 'required', null, 'client');
12. }

```

コード 6-7 「motorskill/edit_motorskill.php」に加筆したソースコード

「motorskill/questiontype.php」は、運動技能の問題を作成したり編集したりするとき
にデータベースにアクセスするファイルである。このファイルに、コード 6-8 に示したも
のを加筆した。これは、コード 6-7 によって表示されたフォームに入力された「scheme」
をコード 6-6 によって作成されたテーブルに格納するためのプログラムである。

```

01. $this->save_app($question);
   ~略~
02. public function save_app($question) {
03. global $DB;
04. $result = new stdClass();
05. $DB->delete_records('question_motorskill_app', array('question' => $question->i
d));
06. $app = new stdClass();
07. $app->question = $question->id;
08. $app->app_id = $question->appid;
09. $DB->insert_record('question_motorskill_app', $app);
10. $result->app = $app;
11. return $result;
12. }

```

コード 6-8 「motorskill/questiontype.php」に加筆したソースコード

「motorskill/renderer.php」は、運動技能の問題を表示するときに呼び出されるファイ
ルである。このファイルに、コード 6-9 に示したものを加筆した。2行目では、ページ上
のフォームへの入力を無効化している。4行目および5行目で、問題作成時に入力された
「scheme」を取得している。7行目、10行目および11行目では、学習者用アプリケーシ
ョンに受け渡すパラメータを取得している。9行目では、学習者用アプリケーションを起
動するためのリンクを生成している。

```

01. // ページ上のフォームへの入力を無効化
02. $inputattributes['readonly'] = 'disabled';
    ~略~
03. // scheme を取得
04. $res = $DB->get_record("question_motorskill_app",array('question' => $id));
05. $app = $res->app_id;
06. // 学習者用アプリケーションに受け渡すパラメータを取得
07. $usageid = $qa->get_usage_id();
08. $questiontext = $question->format_questiontext($qa);
09. $questiontext .= "アプリを使って練習します。ここで直接入力することはできません。 <br />";
10. $tempsesskey = sesskey();
11. $tempattempt = $_GET['attempt'];
12. // 学習者用アプリケーションの起動リンクを生成
13. $questiontext .= "<a href='intent://whatever/#Intent;scheme=".$app.";S.sesskey=".$
    $tempsesskey.";S.attempt=".$tempattempt.";S.usageid=".$usageid.";end'>アプリを起
    動する</a>";

```

コード 6-9 「motorskill/renderer.php」に加筆したソースコード

「motorskill/posttomoodle.php」は、学習者用アプリケーションが算出した得点を Moodle 側で受け取るためのファイルである。このファイルのみ、「数値問題」から引き継いだものではなく、外部(学習者用アプリケーション)との連携のために新規に作成した。ソースコードをコード 6-10 に示す。学習者用アプリケーションが算出した得点「answers」はコンマ続きの文字列として受け渡されるが、3行目から6行目で、学習者用アプリケーションからのパラメータを受け取っている。8行目から20行目は、受け渡された得点の処理部分であり、コード 6-5 で述べたためここでは省略する。22行目から24行目では、ログイン情報等を処理し、29行目から36行目によって、これらのパラメータが、小テストの受験後の処理を行うプログラムである「processattempt.php」に受け渡され、Moodle の標準機能でのフィードバックの提示やデータベースへの格納が行われる。

```

01. <?php
02. // 学習者用アプリケーションから渡されたパラメータの受け取り
03. $answers = $_GET['answers'];
04. $sesskey = $_GET['sesskey'];
05. $attempt = $_GET['attempt'];
06. $usageid = $_GET['usageid'];
07. // 受け取ったコンマ続きの得点を配列に変換
08. $answer_array = explode(",", $answers);
09. $html = "";
10. $slots = "";
11. // 得点等のパラメータをループで処理して整形
12. for ($i=0; $i<count($answer_array); $i++) {
13. $j = $i+1;
14. $html .= "<input type='hidden' name='q".$usageid.":".$j."_:flagged' value='0'>";
15. $html .= "<input type='hidden' name='q".$usageid.":".$j."_:sequencecheck' value='

```

```

1'>";
16. $html .= "<input type='hidden' name='q'.".$usageid."":".$j."_answer' value='".$answer_array[$i]."'>";
17. $slots .= $j;
18. if ($i != count($answer_array)-1) {
19. $slots .= ",";
20. }
21. }
22. $html .= "<input type='hidden' name='attempt' value='".$attempt."'>";
23. $html .= "<input type='hidden' name='sesskey' value='".$sesskey."'>";
24. $html .= "<input type='hidden' name='slots' value='".$slots."'>";
25. ?>
26. <html>
27. <body onload='document.FRM.submit();' method='POST'>
28. // 成形したパラメータを送信
29. <form action='../mod/quiz/processattempt.php' name='FRM'>
30. <?php echo $html; ?>
31. <input type='hidden' name='next' value='次へ'>
32. <input type='hidden' name='thispage' value='0'>
33. <input type='hidden' name='nextpage' value='-1'>
34. <input type='hidden' name='timeup' value='0'>
35. <input type='hidden' name='scrollpos' value=''>
36. </form>
37. </body>
38. </html>

```

コード 6-10 「motorskill/posttomoodle.php」のソースコード

6.3. プラグインを用いた学習コースの例

開発したプラグインを用いて、Moodle 上での学習コースの一例を作成した（表 6-2, 図 6-10）。ここで例として示す学習コースは、以下に示す「1. まずは見てみよう」から「8. やってみよう」の 8 つの学習リソースから成る。「1. まずは見てみよう」から「7. もう一度見てみよう」では Moodle の標準機能を用いて、文字や図、動画で認知段階の学習を支援し、「8. やってみよう」で開発したプラグインにより学習者用アプリケーションと連携し、連合段階および自動化段階の学習支援をするコースである。

表 6-2 学習コースの一例

学習リソース	学習の段階/プラグインの使用
1. まずは見てみよう	} 認知段階/無
2. かんなの握り方	
3. 立ち位置	
4. 動作のポイント1: 引きの速度	
5. 動作のポイント2: かんなのブレの少なさ	
6. 動作のポイント3: 手元と腰の動き出しのタイミング	
7. もう一度見てみよう	
8. やってみよう	} 連合段階・自動化段階/有

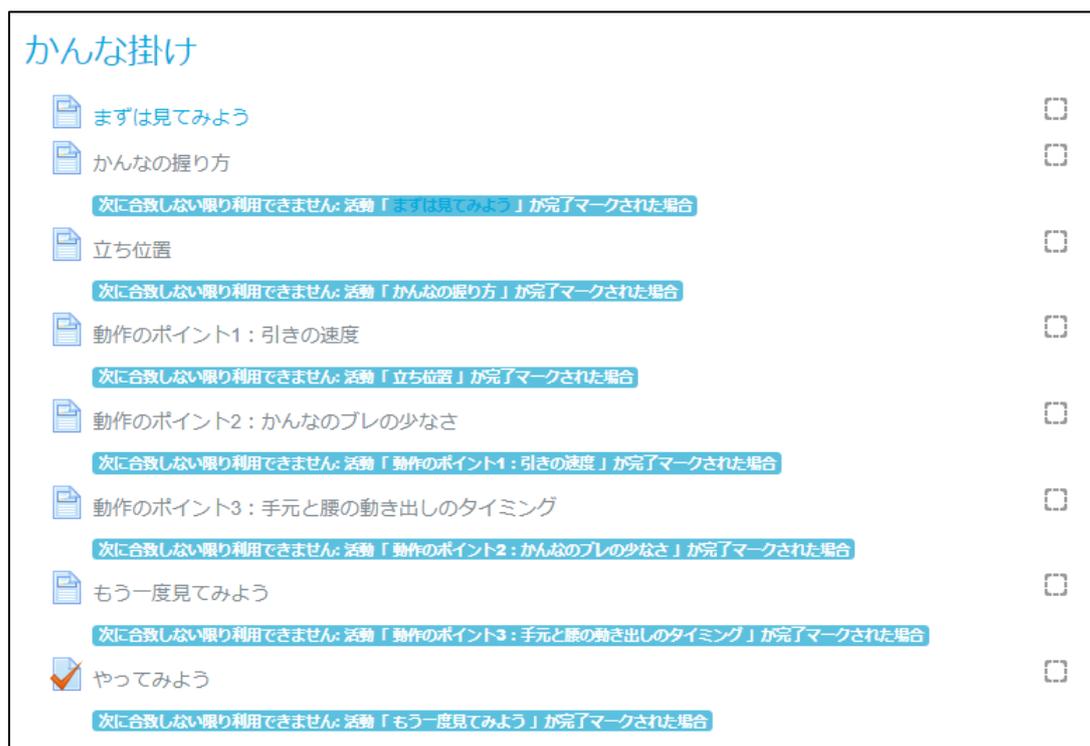


図 6-10 開発したプラグインを用いた学習コースの例

具体的には、はじめに「1. まずは見てみよう」で熟練者のかんな掛けを動画で観察する。次に、「2. かんなの握り方」(図 6-11)や「3. 立ち位置」で、かんな掛けを行う上で前提となる事項を学習する。その後「4. 動作のポイント1: 引きの速度」から「6. 動作のポイント3: 手元と腰の動き出しのタイミング」(図 6-12)で、重要となるポイント

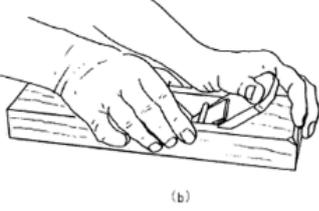
を認知的に学習し、そのポイントを踏まえて、再度熟練者のかんな掛けを動画で観察する。最後に、「8. やってみよう」で実際に学習者自身がかんな掛けを行い、フィードバックを得る。開発したプラグインが支援しているのは、この「8. やってみよう」の段階である。

かんなの握り方 ⚙️

両手を使って握ります。
下の図は右利きの場合の握り方です。

- 右手の平をかんな台の上端（うわば）の中ほどに密着させ、指を小端面にかける
- 左手は台頭とかんな刃にかける

※両手とも、かんなを握った指先は、かんな台の下端（したば）から絶対に出さないように注意しましょう。
出していると、削るときにとげが刺さるなどのけがをします。



●参考・引用
三訂木工工作法
職業能力開発総合大学校 基盤整備センター／編

図 6-11 「2. かんなの握り方」を学習するページ

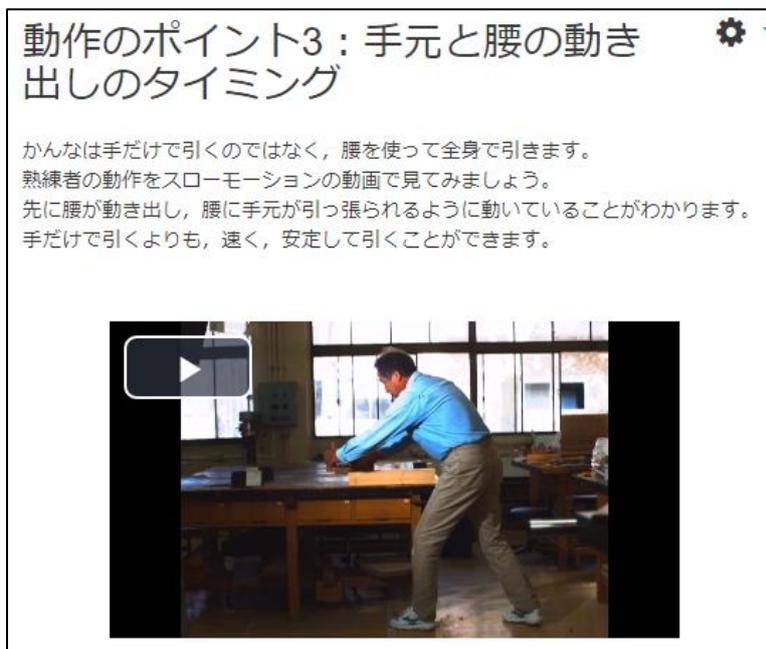


図 6-12 「6. 動作のポイント3: 手元と腰の動き出しのタイミング」を学習するページ

「8. やってみよう」において学習者用アプリケーションを用いて練習を行うと、得点に応じたフィードバックが提示されることは前節で述べた通りであるが、フィードバックの中に提示したい教材へのリンクを埋め込むことにより、学習者を意図した教材へ導くことができる。例えば、図 6-13 は「手元と腰の動き出しのタイミング」の観点で、手元だけを使っていた場合のフィードバックである。ここには、学習コース内の「6. 動作のポイント3: 手元と腰の動き出しのタイミング」や「7. もう一度見てみよう」へのリンクが表示されるように指定した。

この学習コースにおける学習のフローを図にすると図 6-14 のようになる。各学習リソースは、Moodleの標準機能を使用して、1つ前の学習リソースでの学習が完了しないと次の学習リソースへ進めないように設定することができる。また前述の通り、小テストの結果に応じて表示したい学習リソースへ導くことができる。開発したプラグインを用いることにより、e-Learningによる学習においても、認知段階の学習と連合段階および自動化段階の学習を組み合わせることで学習を管理することを実現した。なお、図 6-14 の網掛け部分は、開発したプラグインによって実現した箇所を示す。

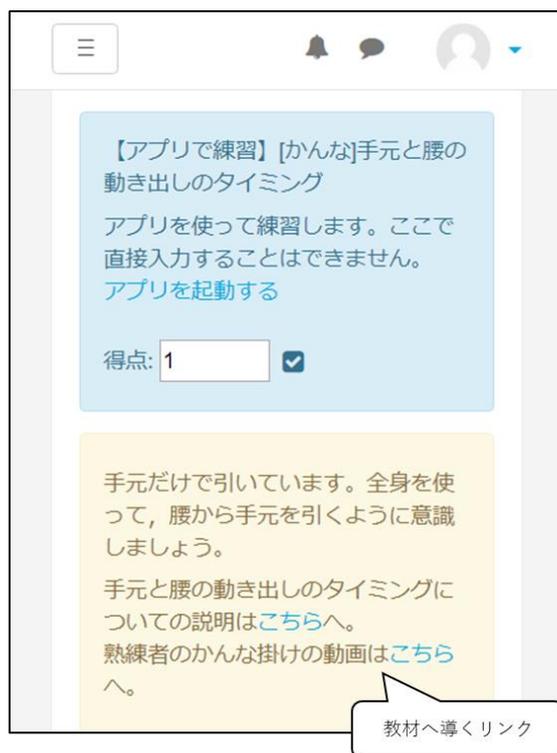


図 6-13 「手元と腰の動き出しのタイミング」におけるフィードバック

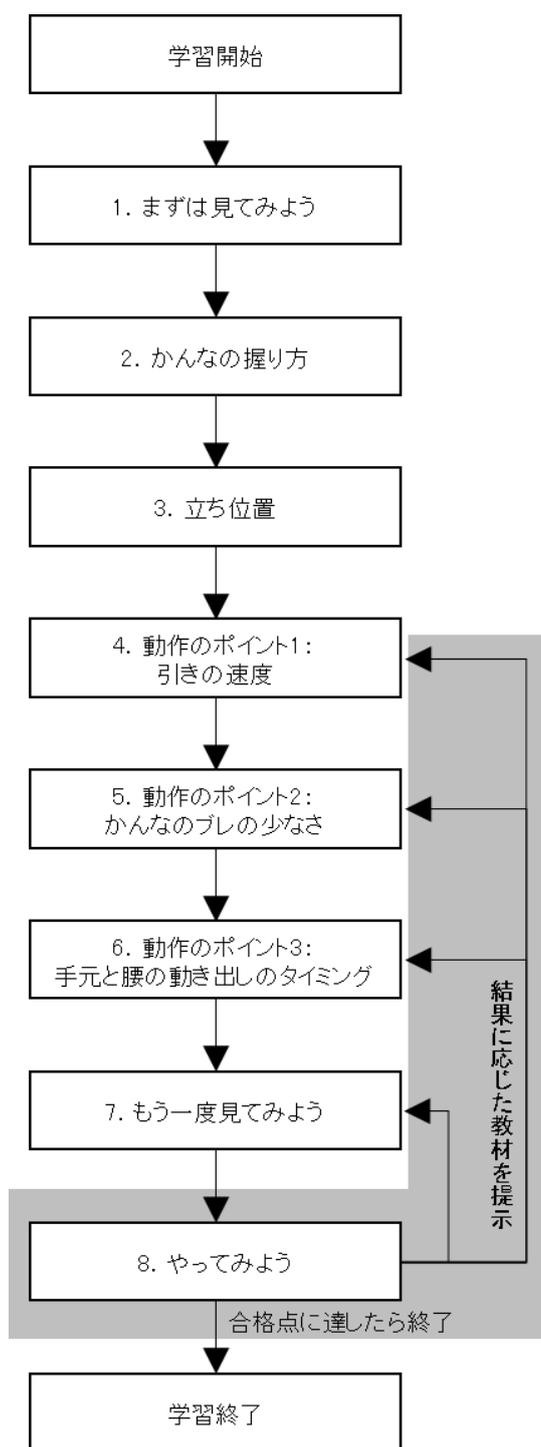


図 6-14 学習コース例における学習のフロー

6.4. 他の運動技能への応用可能性

評価する観点の数が異なる他の運動技能を対象とした場合でも、本プラグインが使用可能であることの検証を行った。他の運動技能としてのこぎり引き動作を対象とし、安藤ほか(2013)が開発したのこぎり引き動作を対象とした学習者用アプリケーションを使用し、本プラグインの期待通りの動作が得られるか確認を行った。

安藤ほか(2013)が開発したのこぎり引き動作を対象とした学習者用アプリケーションは、学習者ののこぎり引き動作を、引きの速度、のこぎりのブレの少なさ、引き込み角度、切断にかかった時間、引いた回数、の5観点で評価し、学習者にフィードバックを与えるものである(図6-15)。

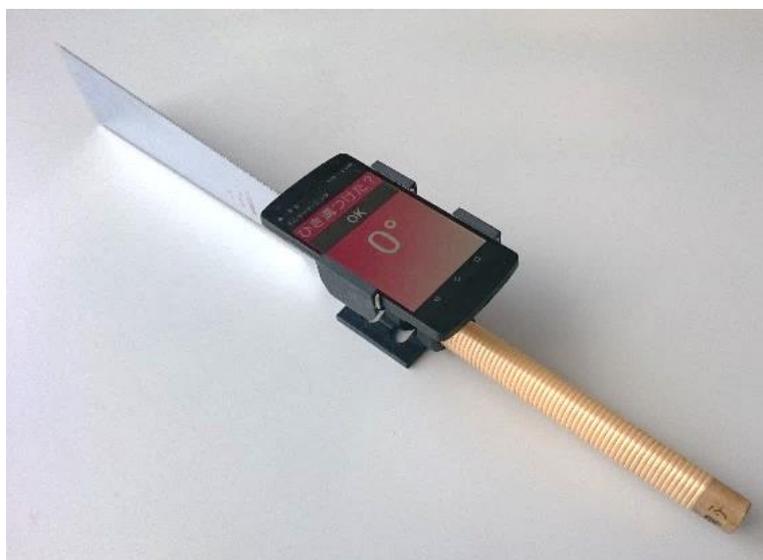


図 6-15 のこぎり引きを対象とした学習者用アプリケーション

このアプリケーションに、前節で述べたものと同様のコーディングを加えた(コード6-11, コード6-12)。また、Moodle上で5観点に対応した問題およびそれを含む小テストを作成した(図6-16)。その上で、小テストを受験する一連の流れを試行した結果、期待通りの動作が得られることを確認した。

以上のことから、開発したプラグインにより、本研究で開発した学習者用アプリケーションだけでなく他のアプリケーションでも、評価する観点の数に関わらず、運動技能のセンシングデータを基に算出された運動技能の得点を、学習履歴としてMoodleで管理することができるようになったといえる。

```
01. // Moodle から起動されたことを確認
02. if (String.valueOf(getIntent().getCategories()).equals("{android.intent.category.
    BROWSABLE}")){
03. // 各パラメータの受け取り
04. sesskey = extras.getString("sesskey");
05. attempt = extras.getString("attempt");
06. usageid = extras.getString("usageid");
07. browser = "com.android.browser";
08. } else {
09. // エラーメッセージの表示
10. Toast.makeText(this, "このアプリは Moodle から起動して使用します. ", Toast.LENGTH_LON
    G).show();
11. finish();
12. }
```

コード 6-11 のこぎり引き動作を対象とした学習者用アプリケーション起動時の
Moodle からのデータの受け取り

```
01. // Moodle 側の受け取り用のプログラムを指定
02. Uri uri =Uri.parse(moodlehost + "question/type/motorskill/posttomoodle.
    php?answers="+answers+"&sesskey="+sesskey+"&attempt="+attempt+"&usageid="+usagei
    d );
03. // 受け渡すパラメータを記述
04. Intent intent = new Intent(Intent.ACTION_VIEW, uri);
05. byte[] postdata = ("answers="+answers+"&sesskey="+sesskey+"&attempt="
    +attempt).getBytes();
06. Intent intent2 = intent.putExtra("com.android.browser.post_data", post data);
07. startActivity(intent2);
```

コード 6-12 のこぎり引き動作を対象とした学習者用アプリケーションが
算出した得点の Moodle への受け渡し



図 6-16 のこぎり引きの 5 観点に対応した小テスト

6.5. まとめと課題

学習者用アプリケーションを用いた学習を汎用の LMS で管理するためのプラグインを開発した。汎用の LMS には Moodle を用いた。開発にあたっては、様々な運動技能に対応できるようにするために、評価する観点の数に関わらず動作するよう設計した。かなな掛けおよびのこぎり引きのそれぞれを対象とした学習者用アプリケーションにおいて、本プラグインを用いて Moodle で学習を管理することができることを確認した。以下の 3 点を満たすことにより、運動技能の学習を汎用の LMS で管理することができることが明らかになったといえる。

- 要件1. 対象とする運動技能におけるいくつかの然るべき観点において運動技能をセンシングし、各観点の得点を算出する学習者用アプリケーションがあること
- 要件2. LMS 上で評価する観点の数に応じた問題を作成できること
- 要件3. 要件 1 で算出された得点が、要件 2 で作成された各問題に対応した解答となるように連携すること

本プラグインにより、運動技能の学習における連合段階および自動化段階の学習を、従来の認知段階の学習と同様に汎用のLMSで管理することを実現できた。すなわち、習得の状況に応じて適切な教材へジャンプさせたり、演習問題の結果や所要時間を蓄積して学習の進捗を管理したりという、汎用のLMSの機能を利用することができるようになった。

ただし、プラグインの開発および期待通りの動作の確認は行ったものの、本プラグインを用いたe-Learningでの学習が成立するのか、学習効果が対面による学習と比較してどの程度見られるのかなど、今後検討されるべき課題が残っている。e-Learningによる一連の学習を行い、効果を評価したい。

第7章 成果と今後の課題

研究の背景では、従来行われてきた運動技能の学習方法では困難点があることや、それに対して、運動技能のセンシングを含む ICT を活用した学習支援が行われていることを挙げた。その上で、従来のセンシングの手法が、高価な装置や大掛かりな手間を必要とするものであり、日常の学習での活用には適しておらず、運動技能の学習における連合段階および自動化段階の学習支援が実現していないことを問題として捉えた。そして、従来のセンシングの手法よりも低いコストで、かつ簡便に運動技能をセンシングし、取得したデータに基づいて運動技能の学習における連合段階および自動化段階の学習を支援する方法を、システム等の開発および評価を通して検討することを目的とした。センシングには身近に普及しているデバイスであるスマートフォンを用いた。対象とした運動技能はかんな掛けであり、授業のような対面による学習と e-Learning による学習の 2 つの視点から検討を行った。

開発したシステム等は、運動技能のセンシングデータを基にフィードバックを提示する学習者用スマートフォンアプリケーション、運動技能のセンシングデータを基に指導参考情報を提示する教師用システム、および運動技能の学習を汎用の LMS で管理するためのプラグインの 3 つであった。

7.1. 本研究の成果

身近に普及しているスマートフォンを用いることにより、従来のセンシングの手法よりも低いコストで、かつ簡便に運動技能のセンシングや、取得したデータに基づいた運動技能を上達させるためのフィードバックの提示を実現したことが本研究において基盤となる成果である。これにより、運動技能の学習における連合段階および自動化段階を対象とした、授業のような対面による学習や e-Learning による学習における、センシングや取得したデータを活用した学習支援の方法を検討することができた。このことを具体的に、以下の 3 つに分けて整理した。

【成果 1】スマートフォンを用いた、従来のセンシングの手法よりも低いコストで、かつ簡便な運動技能のセンシングおよびフィードバックの提示により、運動技能の学習における連合段階および自動化段階の学習支援を実現した

第3章および第5章において、スマートフォンを用いて学習者の運動技能をセンシングし、取得したデータを基に運動技能を上達させるためのフィードバックを提示することが可能であることを、学習者用アプリケーションの開発および評価を通して明らかにした。特に、第3章では、専用かなや削るための木材などの必要なものが全て揃った実践的な学習環境において、第5章では必要なものを全て揃えることが困難である模擬的な学習環境において一定の効果があることが明らかにした。これにより、運動技能の学習における連合段階および自動化段階の学習支援を実現できたといえる。

本研究で開発した学習者用アプリケーションは、学習者のかな掛けを3つの観点でセンシングし、そのデータに基づき、あらかじめ登録された熟練者の動作データと比較し、即時的にフィードバックを提示する機能を持つスマートフォンアプリケーションであった。対象となる運動技能によって、評価する観点やその数は異なるが、その観点において運動技能をセンシングし、熟練者の動作と比較した結果をフィードバックとして提示する手法は、他の運動技能の学習支援を検討する際の参考となり得る知見であると考えられる。

模擬的な学習環境においても一定の効果がみられたことは、場所を選ばず簡便に使用可能であることを示しており、また学習環境の拡張という面で貴重な知見であるといえる。

【成果 2】対面による運動技能の学習における、運動技能のセンシングデータを活用した連合段階および自動化段階の学習支援環境を実現した

第3章および第4章で得られた結果から、対面による学習における、運動技能のセンシングデータを活用した学習支援環境を実現することができたといえる。

学習者用アプリケーションを用いることにより学習者を支援できることは、成果1の通りである。さらに、学習者用アプリケーションにより取得されたセンシングデータを収集し、指導参考情報として提示するシステムがあることで、教師による机間指導を効率化することができた。

学習者にとって、教師から指導を受けることは重要であるが、教師には、多数の学習者を同時に指導しなければならない難しさがある。本研究ではこの難しさを軽減し、指導を効率化することができた。これにより、学習者が充実した指導を受けることができるよう

になった。具体的には、教師から遠い位置にいて目視では確認しづらい学習者の状態の確認や、リアルタイムで確認することができない学習者の状態の確認という点において、センシングしたデータを基に提示された指導参考情報が役立つことが示された。

従来、動画教材等を用いた認知段階の学習支援しかなかった対面による学習において、取得したセンシングしたデータに基づいて、学習者には運動技能を上達させるためのフィードバックを、教師には指導参考情報を提示することによって、運動技能の学習における連合段階および自動化段階の学習が支援可能であることが明らかになった。

【成果3】 e-Learning による運動技能の学習において、従来の認知段階の学習と同様に、連合段階および自動化段階の学習も汎用の LMS で管理することを実現した

第6章では、学習者用アプリケーションにより取得したセンシングデータを基に、汎用の LMS で学習を管理するためのプラグインを開発した。これにより、運動技能の学習における連合段階および自動化段階の学習も従来の認知段階の学習と同様に LMS で管理できるようになった。本プラグインは、対象とする運動技能をセンシングし、各観点の得点を算出する学習者用アプリケーションがあれば、評価する観点の数にかかわらず使用可能となるよう、設計されている。

また、第3章で評価したような実践的な学習環境と LMS を組み合わせた学習も当然可能であるが、第5章で得られた、模擬的な学習環境においても学習が可能であるという知見は、いつでもどこでも学習できるという e-Learning の利点を助長するものであるといえる。

本研究の今日的な意義を述べる。

運動技能のセンシングは高価で大掛かりであるため、日常の学習での活用が困難とされていた。本研究では、運動技能のセンシングを、スマートフォンという比較的安価で身近なデバイスによって実現した。新機種購入に伴って自宅等に放置されている中古のスマートフォンを用いればコストはかからずに済む。

スマートフォンによって運動技能をセンシングし、取得したデータに基づいてフィードバックを提示することにより、運動技能の学習における連合段階および自動化段階の自己学習が実現した。

学校現場における対面による学習においては、個々の学習状況を教師が十分に把握する

ことが困難なために、多数の学習者が一人の教師から十分な指導を受けることができないことが課題とされていたが、スマートフォンによって取得したセンシングデータを教師に提示することにより、教師による指導を効率化することができ、学習者が個に応じた指導を受けることを実現することができた。

また、e-Learning による学習においては、知識に関する学習は広く普及していることに對し、運動技能についての学習環境は十分ではなかったが、スマートフォンによって取得したセンシングデータを基に LMS で学習を管理することにより、e-Learning においても運動技能を学習することが可能となった。加えて、模擬的な学習環境における効果も認められ、生涯学習における学習環境を拡張することができた。

今後、さらに ICT が発展し、ウェアラブルデバイス等のセンサを内蔵した端末や IoT 等が普及すると考えられる。運動技能のセンシングはさらに容易になり、本研究で得られた知見を活用できる場面が増えることが期待される。そのような時代に先立って本研究での知見が得られたことは意義があるといえる。

7.2. 今後の課題

主に4点の課題が挙げられる。

1 点目として、本研究では、対象とする運動技能の一例としてかんな掛けを取り上げたが、その他の運動技能へ応用することが課題として挙げられる。例えば、体育の授業での運動技能や、スポーツ競技の動作などに応用可能であると考えられる。応用するにあたり、各運動技能において評価する観点を定める必要がある。その方法として、本研究のように、「2.5.4.かんな掛けを対象とすることの妥当性」で述べた、Johnson (1961) による運動技能の4つの要素「form」、「accuracy」、「speed」、「adaptability」に基づきながら、先行研究で明らかになっている知見から評価する観点を採った上で、使用する機器（本研究でいうスマートフォン）でセンシング可能な観点到絞る、という手続きを踏むことで、妥当な観点を定めることができると考えられる。

2 点目として、本研究においては、運動技能の学習を汎用の LMS で管理するためのプラグインの開発を行い、運動技能の学習における連合段階および自動化段階の学習も従来の認知段階の e-Learning と同様に学習できる環境を実現したが、その学習効果の評価は行っていない。開発したプラグインを用いた e-Learning での学習が成立するのか、対面

による学習と比較してどの程度の学習効果が見られるのかなど、今後検討されるべき課題が残っている。

3点目として、本研究では、「2.2.運動技能の分類」で述べたように、かんな掛けの被削材のように、その運動技能の結果生まれる成果がある場合でも、身体動作の評価に焦点化しており、成果の評価を対象外としていた。運動技能の結果生まれる成果が存在する場合には、身体動作の評価に加えて成果も併せて評価することができれば、より良い学習支援を提供することができる。かんな掛けにおける成果としては被削材の仕上がりが該当するが、木材の加工物の仕上がりを評価する方法として、福谷ほか（2017）が、スマートフォンを用いて加工物の表面粗さや傾きを評価する機構を提案している。このような評価方法により、身体動作の評価と成果の評価を組み合わせることが実現できると考えられる。

4点目として、本研究ではフィードバックのために画面表示のみを用いた。しかし、スマートフォンのような携帯情報端末は、音声やバイブレーションなど、他にも様々な機能を有している。これらを用いて、例えば引きの速度に応じた切削音を再生したり、バイブレーションにより切削抵抗を疑似的に感じさせたりして、模擬的な学習環境での学習における臨場感を付加するなど、より充実した学習支援を実現できると考えられる。

今後、ICTのさらなる発展に伴い、運動技能のセンシングが一層容易になり、データが膨大に収集され、人工知能と組み合わせることで分析がなされれば、より効果的なフィードバックを提示することも可能になるものと期待される。

謝辞

本学位論文の執筆にあたり、多くの皆様から多大なるご指導・ご支援をいただきました。

本学位論文は、指導教員である堀田龍也教授の多大なるご指導のもと執筆したものです。ここに感謝の意を表します。

博士後期課程に進学した直後、堀田教授はまず、それまで私が見てきた狭い世界から殻を破るようご指導くださいました。研究をすればするほどに自分の無力さを感じ、堀田教授の前で居酒屋で涙を流したときの悔しさは、今も私の原動力の根底にあります。

堀田教授からいただいたご指導は、研究の遂行に関わることだけではありませんでした。ゼミの発足から関わらせていただき、ゼミを作り上げて運営していくことについて教えていただいたこと、大学の教員としての業務をお手伝いさせていただきながら学ばせていただいたこと、他の研究者や企業との共同研究を共にすることで肌で感じさせていただいたこと、講演や出張に随行させていただき、学校現場をはじめとする社会への研究成果の還元を見させていただいたことなど、全てが私の財産となり、本学位論文の執筆につながっております。また、こうした堀田教授の活動を、3年間、ゼミ中でも最も近い立場で見させていただいたことは、とても贅沢でした。ゼミの師匠であることを超え、生き方の師匠としてご指導くださったことに、心より感謝しております。本当にありがとうございました。

本学位論文をご審査くださいました、信州大学の村松浩幸教授、本学の静谷啓樹教授、窪俊一准教授、和田裕一准教授に感謝申し上げます。審査会では私の研究を前向きにご理解いただき、忌憚のなくまた親切なご意見をたくさんいただきました。そのお陰でより良い論文に仕上げることができたと感じています。また、先生方のご指導から、自分もさらに研究力をつけて、互いの研究を高め合う営みに貢献できるようになりたいと、研究者になることへの思いが一層強くなりました。特に、村松教授におかれましては、私が学部生の頃から、学会に参加するたびに私の研究発表を聞きに来てくださり、ご指導くださいました。ここに改めて感謝申し上げます。

日々の研究生活にあたっては、自身の研究活動に専念することができるよう、窪俊一准教授や小野寺香絵氏から多大なるご指導・ご支援をいただきました。ここに感謝申し上げます。

また、同志としてたくさんの励ましをくださったメディア教育論ゼミの皆様には感謝申し上げます。ゼミの皆様には巻き込んでいただいた共同研究の機会は、私の研究上の見識を深め広げる貴重なものであり、それにより本学位論文の執筆も加速しました。ゼミの皆様と顔を合わせることができるおおよそ月に1回の機会は、研究に関わることから日常の悩みまでを全力で議論することができる、私の大好きな時間でした。特に、同時期に学位論文を執筆した佐藤和紀氏には、入学直後から研究のことに留まらないたくさんの方のことを教えていただきました。励まし合いながら共に執筆できたことを本当に嬉しく感じます。

毎年夏に行われたゼミの合宿では、信州大学の村松浩幸教授、福岡工業大学の石塚丈晴教授、奈良教育大学の小柳和喜雄教授、東京学芸大学の高橋純准教授、東京福祉大学の柴田隆史准教授、長崎大学の瀬戸崎典夫准教授、横浜国立大学の脇本健弘准教授、鳴門教育大学の泰山裕講師、東京大学の池尻良平特任講師より、たくさんのご指導をいただいたことに感謝申し上げます。合宿で外部の先生方からいただいたご指導から、自分の研究について立ち止まって考える貴重な機会を与えていただきました。

また、私が学部生時代からの付き合いであった、本学博士後期課程の福谷遼太氏、川田拓氏に感謝申し上げます。学生プロジェクトでの取り組みなど研究面を共にしたほか、日常の息抜きまで、あらゆる場面で寄り添ってくれる優しい後輩をもてたことを嬉しく思います。

開発したシステム等の評価にあたっては、仙台市内の中学校の先生方や生徒の皆様、大学生の皆様にご協力いただきました。貴重な時間を割いていただき、ご協力を快諾してくださったことに感謝申し上げます。

そして、私を学部から修士課程までご指導くださり、博士後期課程進学後にもご指導くださると同時に、温かく見守ってくださった宮城教育大学の安藤明伸准教授に心より感謝申し上げます。本論文に関わる研究は私が宮城教育大学に在籍中に開始したものであり、こうして博士學位論文としてまとめることができたことは、安藤准教授が私に研究の楽しさ

を教えてください，そして宮城教育大学を離れた後も研究を発展させることをお認めくださったからに他なりません．また，私が博士後期課程への進学に意欲をもちつつも自信もてずにいたとき，こちらが相談する前に，それを感じ取ったかのように進学することを勧め，背中を押してくださったことを忘れません．本当にありがとうございました．技術科教育を専門とする研究者の一人として，今後もぜひ研究を共にさせていただき，受けたいご恩や学びを還元していきたい所存でおります．

最後に，本学位論文の執筆に関わる全ての皆様，そして博士後期課程への進学を認め，生活を支えてくださった両親はじめ家族に，改めて感謝申し上げます．

博士後期課程の修了は研究者としての道の始まりであり，本学位論文を執筆するまでに得た学びや与えていただいたご指導を，今後私がどのように生かしていくのかが問われていると感じています．これからも良い研究をし続けていくことで，お世話になった皆様に恩返しをしていくことをここに誓い，謝辞を閉じます．

平成30年1月22日

板垣 翔大

出典

【第1章】

- 大築立志 (1988) 「たくみ」の科学. 朝倉書店
- 山内光哉, 春木豊 (2001) グラフィック学習心理学: 行動と認知. サイエンス社, 東京
- Knapp, B. N. (1963) Skill in Sport. Routledge & Kegan Paul PLC, London
- 武雄靖 (2013) 機械加工作業者の注視行動分析と技能伝承への応用に関する研究. 東京農工大
学博士学位論文
- 山田隆信 (2009) 職人氣質考. 目白大学人文学研究, 5: 27-39
- 嘉戸直樹, 伊藤正憲 (2008) 運動学習はここまでわかった. 関西理学, 8: 49-52
- 安彦忠彦, 新井邦男, 飯長喜一郎, 木原孝博, 児島邦宏, 堀口秀嗣 (1993) 現代学校教育大事
典③. ぎょうせい, 東京
- 間田泰弘, 塩入睦夫, 鶴田敦子, 大竹美登利, 安孫子啓ほか (2011) 技術・家庭科 [技術分野].
開隆堂出版, 東京
- 安藤明伸, 住川泰希 (2012) モーションキャプチャと仮想空間を利用した鋸引き動作観察教材
の開発と機能評価. 日本教育工学会論文誌, 36 (2): 103-110
- 高真紀子, 茂木瑞穂, 菊地恭子, 小野芳明, 高木裕三 (2011) 小児歯科学模型実習における
e-learning (WebCT) 導入に対する評価. 小児歯科学雑誌, 49 (2): 155-164
- 石川淳子, 木村健一, 美馬のゆり (2005) リフレクション活動による調理熟達化支援. 日本デ
ザイン学会研究発表大会概要集, 52: 109
- 佐藤克美, 海賀孝明, 渡部信一 (2010) 舞踊の熟達化を支援するためのモーションキャプチャ
活用. 日本教育工学会論文誌, 34 (Suppl.): 133-136
- 安孫子啓, 安藤明伸, 池田晃一, 寺岡武 (2005) 平かな削りの指導に関する基礎的研究—未
熟練者の切削動作における三次元分析—. 宮城教育大学紀要, 40: 193-199
- 橋爪一治 (2013) 木材切断技能の運動機能解析. 日本産業技術教育学会第 56 回全国大会講演
要旨集: 122
- 高井由佳, 後藤彰彦, 佐藤ひろゆき, 濱田泰以 (2016) 熟練職人の形式知を取り入れた京壁塗
り習熟 e ラーニング教材の構築. 教育システム情報学会誌, 33 (2): 84-93
- 渡部信一 (2007) 日本の「わざ」をデジタルで伝える. 大修館書店, 東京
- 八村広三郎 (2006) 舞踊とモーションキャプチャ: デジタル技術による伝統芸能の記録と解析.

舞踊学, 2006 (29) : 23-26

寺田務 (2015) ウェルネスのための ICT : 6. ウェアラブルセンシングとヘルスケア. 情報処理, 56 (2) : 165-170

田中通義, 篠田功, 山下晃功, 安孫子啓 (1991) かんな削り訓練装置の開発と訓練効果. 日本産業技術教育学会誌, 33 (4) : 251-256

【第2章】

山内光哉, 春木豊 (2001) グラフィック学習心理学 : 行動と認知. サイエンス社, 東京

松田岩男, 近藤充夫 (1969) 運動学習に関する実験的研究 (IV). 体育学研究, 13 (5) : 84

Davies, D. R. (1945) The effect of tuition upon the process of learning a complex motor skill.

Journal of Educational Psychology, 36 (6) : 352-365

谷浩明 (2012) フィードバック・教示と運動学習. 理学療法ジャーナル, 46 (1) : 19-24

NATALIE P. C., STEVE R. P., and TIMOTHY D. L., (2003) Effects of focus of attention depend on golfers' skill. Journal of Sports Sciences, 2003 (21) : 593-600

石黒照朗, 興野悠太郎, 中澤仁, 高汐一紀, 徳田英幸 (2013) 運動学習におけるフィードバック情報操作の効果に関する研究. 研究報告ヒューマンコンピュータインタラクション, 2013-HCI-155 (4) : 1-6

Bilodeau, E. A., Schumsky, I. M., and A. Donald (1959) Some effects of introducing and withdrawing knowledge of results early and late in practice. Journal of Experimental Psychology, 58 (2) : 142-144

Trowbridge, M. H., and Cason, H. (1932) An experimental study of Thorndike's theory of learning. Journal of General Psychology, 7 : 245-260

Greenspoon, J., and Foreman, S. (1956) Effect of delay of knowledge of results on learning a motor task. Journal of Experimental Psychology, 51 (3) 226-228

嘉戸直樹, 伊藤正憲 (2008) 運動学習はここまでわかった. 関西理学, 8 : 49-52

Knapp. B. N. (1963) Skill in Sport. Routledge & Kegan Paul PLC, London

リチャード・A・シュミット (1994) 運動学習とパフォーマンス—理論から実践へ.

安彦忠彦, 新井邦男, 飯長喜一郎, 木原孝博, 児島邦宏, 堀口秀嗣 (1993) 現代学校教育大事典③. ぎょうせい, 東京

東原義訓 (2008) 我が国における学力向上を目指した ICT 活用の系譜, 日本教育工学会論文

誌. 32 (3) : 241-252

中山和彦, 東原義訓 (1986) マイコン・クラスルーム未来の教室 CAI 教育への挑戦. 筑波出版会, 茨城

石塚丈晴, 堀田龍也, 高田浩二, 石原一彦, 小川雅弘, 森谷和浩, 森清子 (2007) 携帯情報端末を活用した校内外における学習の連続性を重視した小学生向けモバイル学習環境の構築と実践. 教育情報研究, 22 (3) : 19-33

文部科学省 (2014) 学びのイノベーション事業実証研究報告書.

http://www.mext.go.jp/b_menu/shingi/chousa/shougai/030/toushin/1346504.htm (参照日 2017.12.03)

e ラーニングコンソシアム (2008) e ラーニング白書. 東京電機大学出版局, 東京

高岡詠子 (2012) ペタ語義 : e ラーニングと教育の相互関係. 情報処理, 53 (3) : 310-315

教育再生実行会議 (2015) 「学び続ける」社会, 全員参加型社会, 地方創生を実現する教育の在り方について (第六次提言).

http://www.kantei.go.jp/jp/singi/kyouikusaisei/pdf/dai6_1.pdf (参照日 2017.12.03)

植野真臣 (2007) 知識社会における e ラーニング. 培風館, 東京

大澤真也, 中西大輔 (2015) e ラーニングは教育を変えるか. 海文堂出版, 東京

赤倉貴子, 柏原昭博 (2016) 教育工学選書Ⅱ第1巻 e ラーニング/e テスティング. ミネルヴァ書房, 京都

中西大輔, 大澤真也, 大西昭夫, 岡田あずさ, 有田真理子 (2015) Moodle 上で小テスト作成を支援する「e 問つく朗」の開発. リメディアル教育研究, 10 (2) : 211-214

横田裕亮, 足立紘亮, 國宗永佳, 新村正明 (2011) 音声・動画レポート利用支援用 LMS プラグインの開発. 電子情報通信学会技術研究報告 ET 教育工学, 111 (273) : 101-104

板垣翔大, 堀田龍也 (2016) 日本教育工学会における e-Learning 関連論文等の動向. 日本教育工学会第 32 回全国大会講演論文集 : 649-650

黒上晴夫, 泰山裕 (2013) 教育目標をデザインする-授業設計のための新しい分類体系-. 北大路書房, 京都

堀田龍也・木原俊行 (2008) 我が国における学力向上を目指した ICT 活用の現状と課題. 日本教育工学会論文誌, 32 (3) : 253-263

佐藤克美, 海賀孝明, 渡部信一 (2010) 舞踊の熟達化を支援するためのモーションキャプチャ活用. 日本教育工学会論文誌, 34 (Suppl.) : 133-136

- 池元茂, 濱田泰以, 高井由佳 (2016) 自動車補修塗装における熟練技術者に見られる動きの特徴. 人間工学, 52 (Suppl.) : 132-133
- 安孫子啓, 安藤明伸, 池田晃一, 寺岡武 (2005) 平かな削りの指導に関する基礎的研究-未熟練者の切削動作における三次元分析-. 宮城教育大学紀要, 40 : 193-199
- 橋爪一治 (2013) 木材切断技能の運動機能解析. 日本産業技術教育学会第 56 回全国大会講演要旨集 : 122
- 高井由佳, 後藤彰彦, 佐藤ひろゆき, 濱田泰以 (2016) 熟練職人の形式知を取り入れた京壁塗り習熟 e ラーニング教材の構築. 教育システム情報学会誌, 33 (2) : 84-93
- 澤井浩子, 渡守武和音, 上野敬介, 小山恵美 (2013) 塗り動作習熟過程の定量的評価方法に関する研究-EOG 時系列変動に着目した解析-. 時間学研究, 6 : 49-59
- 佐藤克美, 安住陽子, 海賀孝明, 渡部信一 (2008) 舞踊教育における簡易式モーションキャプチャの有用性. 情報処理学会研究報告人文科学とコンピュータ (CH), 2008-CH-080 (2) : 9-13
- 八村広三郎 (2006) 舞踊とモーションキャプチャ:デジタル技術による伝統芸能の記録と解析. 舞踊学, 2006 (29) : 23-26
- 総務省 (2017) 平成 29 年度版情報通信白書.
<http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/pdf/> (参照日 2017.12.03)
- 渡辺雄貴, 瀬戸崎典夫, 森田裕介, 加藤浩, 西原明法 (2014) モバイルラーニング動画コンテンツの指示方法に関する一考察. 日本教育工学学会誌, 38 (Suppl.) : 109-112
- 村田和也, 藤本貴之 (2013) スマートフォンの授業利用を実現させつつ“授業目的外使用”を制限させる授業補助環境の構築. 情報処理学会インタラクション, 2013-Interaction (1EXB-34) : 267-269
- 寺田務 (2015) ウェルネスのための ICT : 6. ウェアラブルセンシングとヘルスケア. 情報処理, 56 (2) : 165-170
- 藤沖零士, 田中通義, 向井辰徳 (2013) 平鉋による平面削り指導のための右手の力と左手の力の計測システムの開発と評価. 日本産業技術教育学会誌, 55 (3) : 207-212
- 安孫子啓 (1997) 平かな削りの指導に関する基礎的研究-被削材の樹種及び長さの違いによる切削動作の変化-. 宮城教育大学紀要, 32 : 227-231
- 田中通義, 篠田功, 山下晃功, 安孫子啓 (1991) かな削り訓練装置の開発と訓練効果. 日本産業技術教育学会誌, 33 (4) : 251-256

大工道具研究会（2009）*鉋大全～カンナの使いこなしを網羅した決定版～*. 誠文堂新光社, 東京

文部科学省（2008a）*中学校学習指導要領*. 東山書房, 京都

文部科学省（2008b）*中学校学習指導要領解説 技術・家庭編*. 教育図書, 東京

全日本中学校技術・家庭科研究会（2013）平成 24 年度中学校技術・家庭科に関する全国アンケート調査【技術分野】調査報告書. http://ajgika.ne.jp/doc/2013enquete_g.pdf（参照日 2017.12.03）

安東茂樹, 塩入睦夫, 竹野英敏, 間田泰弘, 大竹美登利ほか（2016）*技術・家庭科 [技術分野]*. 開隆堂出版, 東京

佐竹隆顕, 市川道和, 淡野一郎, 門田和雄, 星野欣也ほか（2016）*新技術・家庭 技術分野*. 教育図書, 東京

田口浩継, 佐藤文子, 金子佳代子, 加藤幸一, 赤崎真弓ほか（2016）*新編 新しい技術・家庭 技術分野*. 東京書籍, 東京

田中通義, 安孫子啓（1990）*かんな削り動作における身体重心の移動分析について*. 日本産業技術教育学会誌, 32（4）: 249-255

橘田紘洋, 近藤文彦, 白井博成（2004）*中学生における効果的なかんな切削動作に関する研究*. 愛知教育大学教育実践総合センター紀要, 7 : 297-303

橘田紘洋, 近藤文彦, 白井博成（2003）*中学生における効果的なかんな切削動作に関する研究 かな台の握り位置が切削動作に及ぼす影響について*. 愛知教育大学研究報告 芸術・保健体育・家政・技術科学・創作編, 52 : 35-39

木材加工教育研究会（1983）*技術・家庭教育講座新訂木材加工*, 開隆堂, 東京

職業能力開発総合大学校能力開発研究センター（2009）*木材加工系実技教科書*, 雇用問題研究会, 東京

H.W.Johnson（1961） $SKILL = SPEED \times ACCURACY \times FORM \times ADAPTABILITY$. *Perceptual and Motor Skills*, 13 : 163-170

【第3章】

田中通義, 篠田功, 山下晃功, 安孫子啓（1991）*かんな削り訓練装置の開発と訓練効果*. 日本産業技術教育学会誌, 33（4）: 251-256

田中通義, 安孫子啓（1990）*かんな削り動作における身体重心の移動分析について*. 日本産業

技術教育学会誌, 32 (4) : 249-255

安孫子啓, 安藤明伸, 池田晃一, 寺岡武 (2005) 平かな削りの指導に関する基礎的研究—未熟練者の切削動作における三次元分析—. 宮城教育大学紀要, 40 : 193-199

中山和彦, 東原義訓 (1986) マイコン・クラスルーム 未来の教室 CAI 教育への挑戦. 筑波出版会, 茨城

石塚丈晴, 堀田龍也, 高田浩二, 石原一彦, 小川雅弘, 森谷和浩, 森清子 (2007) 携帯情報端末を活用した校内外における学習の連続性を重視した小学生向けモバイル学習環境の構築と実践. 教育情報研究, 22 (3) : 19-33

【第4章】

文部科学省 (2012) 「少人数学級の推進など計画的な教職員定数の改善について」～子どもと正面から向き合う教職員体勢の整備～.

http://www.mext.go.jp/a_menu/shotou/hensei/003/_icsFiles/afieldfile/2012/09/18/1325926_1.pdf (参照日 2017.12.03)

河合康則, 瀬尾和哉, 日高貴志夫, 三澤正太 (2014) 山形県における技術科教育の実態と課題に関する調査研究, 日本産業技術教育学会第 32 回東北支部大会講演論文集 : 35-36

中山和彦, 東原義訓 (1986) マイコン・クラスルーム 未来の教室 CAI 教育への挑戦. 筑波出版会, 茨城

石塚丈晴, 堀田龍也, 高田浩二, 石原一彦, 小川雅弘, 森谷和浩, 森清子 (2007) 携帯情報端末を活用した校内外における学習の連続性を重視した小学生向けモバイル学習環境の構築と実践. 教育情報研究, 22 (3) : 19-33

間田泰弘, 塩入睦夫, 鶴田敦子, 大竹美登利, 安孫子啓ほか (2011) 技術・家庭科 [技術分野]. 開隆堂出版, 東京

星野敏彦, 近藤明博, 加藤幸一 (2003) 技術科教育における教具の機能分析及び評価—電気・情報基礎療育の場合—. 日本産業技術教育学会誌, 45 (2) : 91-98

安藤明伸, 住川泰希 (2012) モーションキャプチャと仮想空間を利用した鋸引き動作観察教材の開発と機能評価. 日本教育工学会論文誌, 36 (2) : 103-110

【第5章】

- 佐藤克美, 安住陽子, 海賀孝明, 渡部信一 (2008) 舞踊教育における簡易式モーションキャプチャの有用性. 情報処理学会研究報告人文科学とコンピュータ (CH), 2008-CH-080 (2) : 9-13
- 田中通義, 山崎信寿 (1990) かな削り動作の計測と評価. 人間工学, 26 (Suppl.) : 120-121
- 田中通義, 安孫子啓 (1990) かな削り動作における身体重心の移動分析について. 日本産業技術教育学会誌, 32 (4) : 249-255
- 安孫子啓, 安藤明伸, 池田晃一, 寺岡武 (2005) 平かな削りの指導に関する基礎的研究—未熟練者の切削動作における三次元分析—. 宮城教育大学紀要, 40 : 193-199
- 藤沖零士, 田中通義, 向井辰徳 (2013) 平鉋による平面削り指導のための右手の力と左手の力の計測システムの開発と評価. 日本産業技術教育学会誌, 55 (3) : 207-212
- 近藤義美 (1976) 操作における動作の定位点. 日本産業技術教育学会誌, 18 (1) : 123-127
- 橘田紘洋, 近藤文彦, 白井博成 (2003) 中学生における効果的なかな切削動作に関する研究—かな台の握り位置が切削動作に及ぼす影響について—. 愛知教育大学研究報告 芸術・保健体育・家政・技術科学・創作編, 52 : 35-39

【第6章】

- 高岡詠子 (2012) ペタ語義 : e ラーニングと教育の相互関係. 情報処理, 53 (3) : 310-315
- 植野真臣 (2007) 知識社会における e ラーニング. 培風館, 東京
- 赤倉貴子, 柏原昭博 (2016) 教育工学選書Ⅱ第1巻 e ラーニング/e テスティング. ミネルヴァ書房, 京都
- Moodle Pty Ltd. (2017) About Moodle – MoodleDocs.
https://docs.moodle.org/34/en/About_Moodle (参照日 2017.12.03)
- 嘉戸直樹, 伊藤正憲 (2008) 運動学習はここまでわかった. 関西理学, 8 : 49-52
- 板垣翔大, 堀田龍也 (2016) 日本教育工学会における e-Learning 関連論文等の動向. 日本教育工学会第32回全国大会講演論文集 : 649-650
- 渡辺雄貴 (2015) 新たな教育手法をカリキュラムにどう組み込むか (インストラクショナルデザインの観点から). リメディアル教育研究, 10 (2) : 134-142
- 安藤明伸, 高久敏宏, 板垣翔大, 鳥居隆司, 竹野英敏 (2013) スマートフォンによるセンサーアシストトレーニング法の開発. モバイル学会モバイル'13 研究論文集 : 71-72

【第7章】

H.W.Johnson (1961) SKILL = SPEED × ACCURACY × FORM × ADAPTABILITY.

Perceptual and Motor Skills, 13 : 163-170

福谷遼太, 安藤明伸, 板垣翔大, 高橋秀幸, 木下哲男 (2017) 技術科教育における技能学習管理システムのための木製加工物評価支援機構の提案. 情報処理学会論文誌 コンシューマ・デバイス&システム, 7 (2) : 51-63

本論文に関する研究発表

A. 査読付き論文

- A-1. 板垣翔大, 安藤明伸, 安孫子啓, 堀田龍也 (2016) かんな掛け動作の学習を支援するスマートフォンアプリケーションの開発と家庭学習における有用性の評価. 日本産業技術教育学会誌, 58 (1) : 39-47
- A-2. 板垣翔大, 安藤明伸, 安孫子啓, 堀田龍也 (2017) かんな掛け動作の学習を支援するための授業用システムの開発と評価. 日本教育工学会論文誌, 41 (1) : 65-75
- A-3. Shota Itagaki, Akinobu Ando, Hiraku Abiko and Tatsuya Horita (2018) Development and Assessment of a Class System to Support Learning of Planing Motion. Educational Technology Research, 40 : Accepted

B. 査読付き国際学会発表

- B-1. Shota Itagaki, Akinobu Ando, Toshihiro Takaku, Hidetoshi Takeno and Takashi Torii (2014) Development of a Skill Practice Management System (PMS) for Learning Japanese Traditional Craft Tools by Using Smartphones. Proceedings of World Conference on Educational Multimedia , Hypermedia and Telecommunications, 2014 : 1001-1009
- B-2. Shota Itagaki, Akinobu Ando, Ryota Fukutani and Tatsuya Horita (2017) Prototype of an E-learning System to Learn Motor Skills for a Traditional Japanese Wood-Processing. Proceedings of E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2017 : 788-797

C. 論文(査読なし)

- C-1. 板垣翔大, 安藤明伸 (2015) かんな掛け技能 e-Learning 専用かんなの開発. 宮城教育大学技術科研究報告, 17 : 16-17

D. 研究会報告（査読なし）

- D-1. 板垣翔大, 福谷遼太, 太田あゆみ, 安藤明伸, 堀田龍也 (2016) 木材の加工動作の学習支援システムと切削物仕上がり評価システムを用いた授業実践. 日本教育工学会研究報告集 JSET16-5 : 257-262

E. 学会発表（査読なし）

- E-1. 板垣翔大, 安藤明伸, 安孫子啓 (2012) 複数のスマートフォンを用いたかんながけ動作習得用教材の開発. 日本産業技術教育学会第 18 回技術教育分科会
- E-2. 板垣翔大, 安藤明伸, 高久敏宏, 鳥居隆司, 竹野英敏 (2013) 平鉋削りの身体コーディネイトのずれをスマートフォンで取得する試み. モバイル学会モバイル' 13 研究論文集 : 73-76
- E-3. 板垣翔大, 安藤明伸, 安孫子啓, 高久敏宏, 浅水智也 (2013) 2 台のスマートフォンに対応させた鉋がけ動作習得用アプリケーションの開発. 日本産業技術教育学会第 56 回全国大会講演要旨集 : 51
- E-4. 板垣翔大, 安藤明伸, 高久敏宏, 福谷遼太 (2013) 技能の可視化は MOOTs (Massive Open Online Trainings) を可能にするか. 日本教育工学会第 29 回全国大会講演論文集 : 255-256
- E-5. 板垣翔大, 安藤明伸, 高久敏宏, 安孫子啓 (2013) 鉋の平削り動作の学習履歴管理システムに必要な要件とは何か. 日本産業技術教育学会第 19 回技術教育分科会講演要旨集 : 11-12
- E-6. 板垣翔大, 安藤明伸, 安孫子啓, 堀田龍也 (2015) かんな掛け動作の学習を支援するアプリケーションの開発と家庭学習での利用. 日本産業技術教育学会第 58 回全国大会講演要旨集 : 174
- E-7. 板垣翔大, 安藤明伸, 安孫子啓, 堀田龍也 (2015) スマートフォンを用いたかんな掛け動作の学習を支援するシステムの開発と授業利用時におけるその有用性の評価. 日本教育工学会第 31 回全国大会講演論文集 : 733-734
- E-8. 板垣翔大, 安藤明伸, 安孫子啓, 堀田龍也 (2016) かんな掛け動作の学習を支援するシステムを用いた授業における授業方法の違いによる学習効果への影響. 日本産業技術教育学会第 59 回全国大会講演要旨集 : 98

- E-9. 板垣翔大, 堀田龍也 (2016) 日本教育工学会における e-Learning 関連論文等の動向.
日本教育工学会第 32 回全国大会講演論文集 : 649-650
- E-10. 板垣翔大, 安藤明伸, 福谷遼太, 堀田龍也 (2017) かんな掛け動作の学習履歴を LMS
で管理するためのプラグインの試作. 日本産業技術教育学会第 60 回全国大会講演要
旨集 : 170

付録

付録 1：授業実践（第 4 章）で用いた質問紙

授業振り返りシート	1 年 組 番 男・女	※氏名は不要です
※この振り返りシートは授業をより良くするためのものですので、率直に書いてください。 ※成績とは関係ありません。		
スマートフォンを使ってかんながけの練習をして感じたことについて、あなたの考えに最も当てはまると思うところに○を付けてください。		
いいえ ←→ はい		
1	かんなを使って木材を削ることは楽しいと思えましたか？	1-2-3-4-5
2	もっと長く練習したいと思えましたか？	1-2-3-4-5
3	意欲的に取り組むことができましたか？	1-2-3-4-5
4	正しいかんなの使い方を理解できましたか？	1-2-3-4-5
5	正しいからだ全身の使い方を理解できましたか？	1-2-3-4-5
6	よく考えて作業できるようになりましたか？	1-2-3-4-5
7	落ち着いた態度で作業をすることができましたか？	1-2-3-4-5
8	先生の説明をしっかりと聞けたと思いますか？	1-2-3-4-5
9	友達と余分な話をしないで授業を受けましたか？	1-2-3-4-5
10	かんながけについて、もっと色々なことを知りたいと思えましたか？	1-2-3-4-5
11	今日の授業で技能が身に付いたと思いますか？	1-2-3-4-5
12	今日のかんながけの練習を作品の製作に活かしていけると思いますか？	1-2-3-4-5
13	操作や使い方がわからなくなることがありましたか？	1-2-3-4-5
14	練習するときに嫌になることがありましたか？	1-2-3-4-5
15	練習するときに困ることがありましたか？	1-2-3-4-5
16	授業の中で何をするかわからなくなることがありましたか？	1-2-3-4-5
17	授業中に「ボーっ」とすることがありましたか？	1-2-3-4-5
18	授業にもっと集中できるとよかったなと思いますか？	1-2-3-4-5
19	練習結果のアドバイスを読み、グラフやアニメーションを見ましたか？	1-2-3-4-5
20	上手に削れるようになろうと意識して練習しましたか？	1-2-3-4-5
21	あなたは、かんなを使って木材の表面を削る加工をしたいと思えますか？	1-2-3-4-5
22	かんなを使って木材の表面をきれいにする自信はありますか？	1-2-3-4-5
23	かんながけの練習をして身に付けた知識や技能が今後の生活に役立つと思えますか？	1-2-3-4-5
24	スマートフォンやアプリを使ったことで、練習へのやる気が高まりましたか？	1-2-3-4-5
25	授業の感想を書いてください。スペースが足りない場合は裏面を使ってください。	

付録 2 : 開発したシステム等の主要なソースコード

2-1 学習者用アプリケーション (教師用システムとの連携時, かなな端末のメインのプログラム)

```
1. package <packageName>;
2. import android.app.Activity;
3. import android.app.AlertDialog;
4. import android.app.ProgressDialog;
5. import android.bluetooth.BluetoothAdapter;
6. import android.bluetooth.BluetoothDevice;
7. import android.bluetooth.BluetoothServerSocket;
8. import android.bluetooth.BluetoothSocket;
9. import android.content.Context;
10. import android.content.DialogInterface;
11. import android.content.SharedPreferences;
12. import android.content.SharedPreferences.Editor;
13. import android.graphics.Bitmap;
14. import android.graphics.Bitmap.CompressFormat;
15. import android.graphics.BitmapFactory;
16. import android.graphics.Canvas;
17. import android.graphics.Color;
18. import android.graphics.Paint;
19. import android.graphics.Path;
20. import android.graphics.PorterDuff;
21. import android.hardware.Sensor;
22. import android.hardware.SensorEvent;
23. import android.hardware.SensorEventListener;
24. import android.hardware.SensorManager;
25. import android.media.MediaPlayer;
26. import android.os.Bundle;
27. import android.os.Environment;
28. import android.os.Handler;
29. import android.os.Message;
30. import android.text.format.Time;
31. import android.util.Log;
32. import android.view.Gravity;
33. import android.view.KeyEvent;
34. import android.view.LayoutInflater;
35. import android.view.View;
36. import android.view.View.OnClickListener;
37. import android.view.ViewGroup;
38. import android.webkit.WebView;
39. import android.widget.AdapterView;
40. import android.widget.AdapterView.OnItemClickListener;
41. import android.widget.AdapterView.OnItemClickListener;
42. import android.widget.AdapterView.OnItemClickListener;
43. import android.widget.AdapterView.OnItemClickListener;
44. import android.widget.AdapterView.OnItemClickListener;
45. import android.widget.AdapterView.OnItemClickListener;
46. import android.widget.AdapterView.OnItemClickListener;
47. import android.widget.AdapterView.OnItemClickListener;
48. import android.widget.AdapterView.OnItemClickListener;
49. import android.widget.AdapterView.OnItemClickListener;
50. import org.apache.http.HttpResponse;
51. import org.apache.http.client.methods.HttpGet;
52. import org.apache.http.impl.client.DefaultHttpClient;
53. import org.apache.http.util.EntityUtils;
54. import java.io.BufferedReader;
55. import java.io.BufferedWriter;
56. import java.io.File;
57. import java.io.FileNotFoundException;
58. import java.io.FileOutputStream;
59. import java.io.FileReader;
60. import java.io.IOException;
61. import java.io.InputStream;
62. import java.io.OutputStream;
63. import java.io.OutputStreamWriter;
64. import java.io.UnsupportedEncodingException;
65. import java.math.BigDecimal;
66. import java.util.ArrayList;
67. import java.util.Calendar;
68. import java.util.Set;
69. import java.util.UUID;
70. import java.util.Vector;
71. public class Kanna2Activity extends Activity implements SensorEventListener {
72.     private float y_sensor;
73.     private float x_sensor;
74.     private float goukei;
75.     private float goukei_x;
76.     private String str_rank;
77.     private float y_SensorValue;
78.     private float x_kosei;
79.     private float x_button;
80.     private float x_kosei2;
81.     private int maxInt = 0;
82.     private int minInt = 0;
83.     private int avarageInt_kanna;
84.     private float avarage_koshi;
85.     private int divInt_kanna;
86.     private float div_koshi;
87.     private boolean divflag_kanna = false;
88.     private boolean divflag_koshi = false;
89.     private boolean startFlag;
90.     private SharedPreferences modelpref;
91.     private long start;
92.     private long stop;
93.     private long alltime;
94.     private File file;
95.     private File fileHosei;
96.     private Runnable loop;
97.     private Time time;
98.     private String date;
99.     private String date2;
100.    private int toggle;
101.    private int point;
102.    private int[] currentlyInt_kanna;
103.    private float[] currently_koshi;
104.    private int timesInt;
105.    private int REPEAT_INTERVAL;
106.    private boolean isRepeat;
107.    private Thread thread;
108.    private int wide;
109.    private int hight;
110.    private Bitmap bmp = null;
111.    private Bitmap bmpA = null;
112.    private Bitmap bmp_model = null;
113.    private Bitmap bmp_modelA = null;
114.    private Canvas bmpCanvas;
115.    private Canvas bmpCanvasA;
116.    private TextView textV;
117.    private TextView countTextV;
118.    private TextView userTextV;
119.    private Button countBtn;
```

```

120. private Button changeUser;
121. private LinearLayout countLayout;
122. private float mYOffset;
123. private float mYOffsetA;
124. private MediaPlayer mp = null;
125. Handler mHandler = new Handler();
126. Handler dHandler = new Handler();
127. SensorCalibration SC = new SensorCalibrat
ion();
128. String modeldata_str;
129. String strint;
130. String modelName;
131. String logsttr;
132. int jikan;
133. private Thread threads;
134. String result;
135. boolean finishflg = false;
136. private SensorManager sensorManager;
137. float gyro_value[];
138. boolean isKoshiMove = false;
139. boolean isKannaMove = false;
140. boolean graphKannaMove = false;
141. int modelZureInt;
142. int zureInt;
143. float modelBure;
144. Paint paintSensor = new Paint();
145. Paint paintLine = new Paint();
146. Paint paintLine2 = new Paint();
147. Paint paintBorder = new Paint();
148. Paint paintText = new Paint();
149. BluetoothAdapter mAdapter;
150. Vector<BluetoothDevice> mDevices;
151. final String NAME = "BLUETOOTH_SAMPLE";
152. UUID MY_UUID = UUID.fromString("111111-0
000-1000-1111-00AAECCAFF");
153. ServerThread serverThread;
154. ConnectedThread[] connection = new Connec
tedThread[1];
155. String Data = null;
156. BluetoothSocket[] socket = new BluetoothS
ocket[1];
157. final int KOSHI = 0;
158. final int KANNA = 1;
159. ArrayList<Integer> docchi = new ArrayList
<Integer>();
160. boolean koshiPlot = false;
161. LinearLayout btnLayout;
162. Button okBtn;
163. Button endBtn;
164. Button replayBtn;
165. private float xGraph = 20;
166. private float pastY = 200;
167. private float pastYs = 200;
168. private float pastYA = 200;
169. float gyroValue;
170. ArrayList<Float> bureList = new ArrayList
<Float>();
171. float bureBorder1 = 70;
172. float bureBorder2 = 90;
173. float speedBorder1 = 70;
174. float speedBorder2 = 90;
175. int intCount = 4;
176. String timing;
177. long kanna_time = 0;
178. long koshi_time = 0;
179. long koshi_time_int = 0;
180. float docchif = 0;
181. Calendar cld;
182. String filename;
183. String filenameHosei;
184. String saveText;
185. String saveHosei;
186. String saveKoshi;
187. String saveTime;
188. String saveBure;
189. int practiceNo = 1;
190. AlertDialog.Builder alertDialogBuilder;
191. String username;
192. ArrayList<String> arrayName;
193. ArrayList<String> arrayGakuban;
194. ArrayList<String> arrayKeyId;
195. static ArrayAdapter<String> adapter;
196. Editor e;
197. android.app.AlertDialog.Builder dialog =
null;
198. AlertDialog dialog2;
199. ArrayList<Float> speedList = new ArrayLis
t<Float>();
200. ArrayList<Float> blurList = new ArrayList
<Float>();
201. ArrayList<Float> koshiList = new ArrayLis
t<Float>();
202. ArrayList<String> saveList = new ArrayLis
t<String>();
203. float postSpeed;
204. float postBlur;
205. int speedRank;
206. int blurRank;
207. final int A = 0;
208. final int B = 1;
209. final int C = 2;
210. String speedRankStr;
211. float[] acc = new float[3];
212. float[] gyr = new float[3];
213. long unixTime;
214. long t;
215. float y_hosei;
216. float z_hosei;
217. float Acc;
218. float AccKako;
219. float Spa;
220. float Vel;
221. float kakoVel;
222. float hosei = (float) Math.cos(Math.PI /
4f);
223. int kessonchi = 0;
224. Path path = new Path();
225. Path pathA = new Path();
226. float moveLine = 3;
227. int graphDownReset = 5;
228. int graphDownInt = graphDownReset;
229. ArrayList<Float> Ylist = new ArrayList<Fl
oat>();
230. ArrayList<Float> YlistA = new ArrayList<F
loat>();
231. ArrayList<Float> YlistA2 = new ArrayList<
Float>();
232. ArrayList<Float> Ylist2 = new ArrayList<F
loat>();
233. int XYnum = 0;
234. VgraphView VgraphView;
235. AgraphView AgraphView;
236. AlertDialog.Builder replayDlg;
237. ReplayView replayView;
238. int replayHeight = 0;
239. int replayWidth = 0;
240. float replayRight = 0;
241. float replayCenter = 0;
242. float replayLeft = 0;
243. float replayLength = 0;
244. float replayRatio = 0;
245. float replayPosition = 0;
246. float model_replayPosition = 0;
247. Bitmap kanna;
248. Button replayBtn2;
249. Runnable replayer;

```

```

250. Thread replay;
251. ArrayList<Float> model_Ylist = new ArrayL
    ist<Float>();
252. View resultView;
253. String ViewStr;
254. int userClassNum = 2;
255. String host = "<host>";
256. String ID = "";
257. int speedPoint = 0;
258. int burePoint = 0;
259. int koshiPoint = 0;
260. int totalPoint = 0;
261. int userId;
262. float span = 0;
263. ProgressDialog progressDialog;
264. @Override
265. public void onCreate(Bundle savedInstanceState) {
266.     super.onCreate(savedInstanceState);
267.     sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
268.     gyro_value = new float[4];
269.     AlertDialog.Builder a = new AlertDialog.Builder(this);
270.     a.setMessage("モデルが選択されていません");
271.     a.setPositiveButton("OK", new DialogInterface.OnClickListener() {
272.         public void onClick(DialogInterface dialog, int id) {
273.             finish();
274.         }
275.     });
276.     e = getSharedPreferences("2modelpref",
277.         MODE_WORLD_READABLE | MODE_WORLD_WRITEABLE).edit();
278.     e.putString("username", "");
279.     e.putString("userid", "");
280.     e.commit();
281.     textV = new TextView(this);
282.     textV.setText("ここに結果が表示されます");
283.     VgraphView = new VgraphView(this);
284.     AgraphView = new AgraphView(this);
285.     btnLayout = new LinearLayout(this);
286.     btnLayout.setGravity(Gravity.BOTTOM | Gravity.CENTER_HORIZONTAL);
287.     okBtn = new Button(this);
288.     endBtn = new Button(this);
289.     replayBtn = new Button(this);
290.     countLayout = new LinearLayout(this);
291.     countLayout.setOrientation(LinearLayout.VERTICAL);
292.     countLayout.setBackgroundResource(R.drawable.start);
293.     setContentView(countLayout);
294.     countTextV = new TextView(this);
295.     countTextV.setText("↓");
296.     countTextV.setTextColor(Color.WHITE);
297.     countTextV.setTextSize(250);
298.     countTextV.setGravity(Gravity.CENTER | Gravity.CENTER);
299.     countBtn = new Button(this);
300.     countBtn.setEnabled(false);
301.     countBtn.setBackgroundResource(R.drawable.go_xml);
302.     userTextV = new TextView(this);
303.     changeUser = new Button(this);
304.     changeUser.setText("学習者の変更");
305.     changeUser.setEnabled(false);
306.     countLayout.addView(countBtn, new LinearLayout.LayoutParams(
307.         ViewGroup.LayoutParams.FILL_PARENT, 250));
308.     LinearLayout.LayoutParams lp = new Linear
    Layout.LayoutParams(ViewGroup.LayoutParams.FILL_PARENT,
    ViewGroup.LayoutParams.FILL_PARENT);
    lp.weight = 1f;
    countLayout.addView(countTextV, lp);
    lp = new LinearLayout.LayoutParams(ViewGroup.LayoutParams.FILL_PARENT,
    ViewGroup.LayoutParams.FILL_PARENT);
    countLayout.addView(userTextV);
    userTextV.setGravity(Gravity.CENTER_HORIZONTAL);
    countLayout.addView(changeUser);
    paintSensor.setColor(Color.CYAN);
    paintSensor.setStyle(Paint.Style.STROKE);
    paintSensor.setStrokeWidth(4);
    paintSensor.setAntiAlias(true);
    paintLine.setColor(Color.WHITE);
    paintLine.setStyle(Paint.Style.FILL);
    paintLine.setStrokeWidth(3);
    paintLine2.setColor(Color.WHITE);
    paintLine2.setStyle(Paint.Style.FILL);
    paintLine2.setStrokeWidth(1);
    paintBorder.setColor(Color.WHITE);
    paintBorder.setStyle(Paint.Style.FILL);
    paintBorder.setStrokeWidth(30);
    paintText.setColor(Color.CYAN);
    paintText.setStyle(Paint.Style.FILL);
    paintText.setStrokeWidth(2);
    paintText.setTextSize(30);
    replayView = new ReplayView(this);
    kanna = BitmapFactory.decodeResource(getResources(), R.drawable.kanna);
    LayoutInflater inflater = LayoutInflater.from(this);
    resultView = inflater.inflate(R.layout.result_view, null);
    TabHost host = (TabHost) resultView.findViewById(R.id.tabhost);
    host.setup();
    TabSpec tabComment = host.newTabSpec("Comment");
    tabComment.setIndicator("コメント");
    tabComment.setContent(R.id.resultComment);
    host.addTab(tabComment);
    TabSpec tabGraph = host.newTabSpec("Graph");
    tabGraph.setIndicator("グラフ");
    tabGraph.setContent(R.id.resultGraph);
    host.addTab(tabGraph);
    TabSpec tabReplay = host.newTabSpec("Replay");
    tabReplay.setIndicator("リプレイ");
    tabReplay.setContent(R.id.resultReplay);
    host.addTab(tabReplay);
    countBtn.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
    graphKannaMove = true;
    mp = MediaPlayer.create(Kanna2Activity.this, R.raw.button);
    try {
    mp.prepare();
    } catch (IllegalStateException e) {
    e.printStackTrace();
    } catch (IOException e) {
    e.printStackTrace();
    }
    }
    mp.start();
    mp = MediaPlayer.create(Kanna2Activity.this, R.raw.count);
    try {
    mp.prepare();

```

```

367. } catch (IllegalStateException e) {
368. e.printStackTrace();
369. } catch (IOException e) {
370. e.printStackTrace();
371. }
372. mp.start();
373. CountDown();
374. countBtn.setVisibility(View.GONE);
375. changeUser.setVisibility(View.GONE);
376. countTextV.setTextSize(400);
377. }
378. });
379. changeUser.setOnClickListener(new View.OnClickListener() {
380. public void onClick(View v) {
381. getuser();
382. }
383. });
384. modelpref = getSharedPreferences("2modelpref",
MODE_WORLD_READABLE
385. | MODE_WORLD_WRITEABLE);
386. modeldata_str = modelpref.getString("2modelint", "");
387. strint = modelpref.getString("2modelint",
"");
388. modelName = modelpref.getString("2name",
"");
389. if (modelName != null) {
390. if (!modelName.equals("")) {
391. modelZureInt = Integer.parseInt(modelpref.
getString(modelName
392. + "2zure", ""));
393. logstr = modelpref.getString(modelName +
"2maxInt", "");
394. modelBure = Marumeru(
395. Float.parseFloat(modelpref.getString(mode
lName
396. + "2maxBure", "")), 2);
397. username = modelpref.getString("username",
"");
398. BufferedReader br = null;
399. StringBuilder sb = new StringBuilder();
400. try {
401. try {
402. br = new BufferedReader(new FileReader(En
vironment
403. .getExternalStorageDirectory().getPath()
404. + "/"
405. + getPackageName()
406. + "/two/"
407. + modelName
408. + ".txt"));
409. String str;
410. while ((str = br.readLine()) != null) {
411. sb.append(str);
412. }
413. } finally {
414. if (br != null) {
415. br.close();
416. }
417. }
418. } catch (IOException e) {
419. }
420. String allData = sb.toString();
421. String rawData[] = allData.split(",");
422. for (int i = 0; i < rawData.length; i++) {
423. try {
424. model_Ylist.add(Float.parseFloat(rawData
[i]));
425. replayLength += Float.parseFloat(rawData
[i]);
426. } catch (NumberFormatException e1) {
427. model_Ylist.add(0f);
428. }
429. }
430. replayRatio = (replayHeight - 50) / repla
yLength;
431. mDevices = new Vector<BluetoothDevice>();
432. mAdapter = BluetoothAdapter.getDefaultAda
pter();
433. Set<BluetoothDevice> devices = mAdapter.g
etBondedDevices();
434. for (BluetoothDevice device : devices) {
435. mDevices.add(device);
436. }
437. serverThread = new ServerThread();
438. serverThread.start();
439. } else {
440. a.show();
441. }
442. } else {
443. a.show();
444. }
445. }
446. public void classView() {
447. final String[] classStr = { "1", "2", "3",
"4", "5", "6", "7", "8", "9", "10", "11
"};
448. ListView LV = new ListView(this);
449. adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, clas
sStr);
450. dialog = new AlertDialog.Builder(this);
451. dialog2 = dialog.create();
452. dialog2.setCanceledOnTouchOutside(false);
453. dialog2.setTitle("あなたのクラスを選択して
ください");
454. dialog2.setView(LV);
455. dialog2.show();
456. LV.setAdapter(adapter);
457. LV.setOnItemClickListener(new OnItemClick
Listener() {
458. public void onItemClick(AdapterView<?> it
ems, View view,
459. int position, long id) {
460. userClassNum = Integer.parseInt(classStr
[position]);
461. dialog2.dismiss();
462. getuser();
463. }
464. });
465. }
466. }
467. Runnable countRunnable;
468. public void CountDown() {
469. final int REPEAT_INTERVAL = 1000;
470. final Handler handler = new Handler();
471. countRunnable = new Runnable() {
472. public void run() {
473. intCount--;
474. countTextV.setText(String.valueOf(intCoun
t));
475. if (intCount == 0) {
476. handler.removeCallbacks(countRunnable);
477. countTextV.setTextSize(150);
478. countTextV.setText("始めぬ!");
479. onBtn();
480. } else {
481. handler.postDelayed(this, REPEAT_INTERVA
L);
482. }
483. }
484. };
485. handler.postDelayed(countRunnable, REPEAT
_INTERVAL);
486. }
487. public void onBtn() {

```

```

488.   currentlyInt_kanna = new int[10];
489.   currently_koshi = new float[10];
490.   mHandler = new Handler();
491.   startFlag = true;
492.   isRepeat = false;
493.   REPEAT_INTERVAL = 5;
494.   time = new Time("Asia/Tokyo");
495.   looper = new Runnable() {
496.   public void run() {
497.   while (isRepeat) {
498.   try {
499.   Thread.sleep(REPEAT_INTERVAL);
500.   } catch (InterruptedException e) {
501.   }
502.   mHandler.post(new Runnable() {
503.   public void run() {
504.   doSomething();
505.   }
506.   });
507.   }
508.   }
509.   };
510.   isRepeat = true;
511.   goukei = 0;
512.   goukei_x = 0;
513.   start = 0;
514.   stop = 0;
515.   maxInt = 0;
516.   minInt = 0;
517.   timesInt = 0;
518.   start = System.currentTimeMillis();
519.   time.setToNow();
520.   thread = new Thread(looper);
521.   thread.start();
522.   textV.setTextSize(75);
523.   textV.setTextColor(Color.WHITE);
524.   textV.setText("計測中...");
525.   textV.setPadding(0, 100, 0, 0);
526.   textV.setGravity(Gravity.CENTER);
527.   float y_btn = y_sensor * 10;
528.   y_btn = Math.round(y_btn);
529.   y_btn = y_btn / 10;
530.   SC.setBtnSensorValue(y_btn);
531.   x_kosei = x_sensor * 10;
532.   x_kosei = Math.round(x_kosei);
533.   x_kosei = x_kosei / 10;
534.   x_button = x_kosei;
535.   }
536.   private void doSomething() {
537.   jikan++;
538.   gyro_value[3] = gyro_value[3] + Math.abs(
539.   gyro_value[0])
540.   + Math.abs(gyro_value[1]) + Math.abs(gyro
541.   _value[2]);
542.   int finishjikan = jikan / 10;
543.   y_SensorValue = SC.setNowSensorValue(y_sen
544.   sor);
545.   x_kosei = x_sensor * 10;
546.   x_kosei = Math.round(x_kosei);
547.   x_kosei = x_kosei / 10;
548.   if (startFlag == true) {
549.   timesInt++;
550.   for (int i = 9; i >= 1; i--) {
551.   currentlyInt_kanna[i] = currentlyInt_kann
552.   a[i - 1];
553.   currently_koshi[i] = currently_koshi[i -
554.   1];
555.   currentlyInt_kanna[0] = (int) y_SensorVal
556.   ue;
557.   currently_koshi[0] = Float.parseFloat(Dat
558.   a);
559.   avarageInt_kanna = (currentlyInt_kanna[0]
560.   + currentlyInt_kanna[1]
561.   + currentlyInt_kanna[2] + currentlyInt_ka
562.   nna[3]
563.   + currentlyInt_kanna[4] + currentlyInt_k
564.   nna[5]
565.   + currentlyInt_kanna[6] + currentlyInt_k
566.   nna[7]
567.   + currentlyInt_kanna[8] + currentlyInt_k
568.   nna[9]) / 10;
569.   avarage_koshi = (currently_koshi[0] + cur
570.   rently_koshi[1]
571.   + currently_koshi[2] + currently_koshi[3]
572.   + currently_koshi[4] + currently_koshi[5]
573.   + currently_koshi[6] + currently_koshi[7]
574.   + currently_koshi[8] + currently_koshi
575.   [9]) / 10;
576.   int[] deviationInt_kanna = new int[10];
577.   float[] deviation_koshi = new float[10];
578.   for (int i = 0; i <= 9; i++) {
579.   deviationInt_kanna[i] = Math.abs(currentl
580.   yInt_kanna[i]
581.   - avarageInt_kanna);
582.   deviation_koshi[i] = Math.abs(currently_k
583.   oshi[i]
584.   - avarage_koshi);
585.   }
586.   divInt_kanna = (deviationInt_kanna[0] + d
587.   eviationInt_kanna[1]
588.   + deviationInt_kanna[2] + deviationInt_ka
589.   nna[3]
590.   + deviationInt_kanna[4] + deviationInt_ka
591.   nna[5]
592.   + deviationInt_kanna[6] + deviationInt_ka
593.   nna[7]
594.   + deviationInt_kanna[8] + deviationInt_ka
595.   nna[9]) / 10;
596.   div_koshi = (deviation_koshi[0] + deviati
597.   on_koshi[1]
598.   + deviation_koshi[2] + deviation_koshi[3]
599.   + deviation_koshi[4] + deviation_koshi[5]
600.   + deviation_koshi[6] + deviation_koshi[7]
601.   + deviation_koshi[8] + deviation_koshi
602.   [9]) / 10;
603.   if (divflag_koshi == false && div_koshi >
604.   0.7f) {
605.   koshi_time = System.currentTimeMillis()-7
606.   5;
607.   divflag_koshi = true;
608.   koshiPlot = true;
609.   docchi.add(KOSHI);
610.   koshi_time_int = jikan;
611.   }
612.   if (divflag_kanna == false && divInt_kann
613.   a > 0) {
614.   divflag_kanna = true;
615.   isKannaMove = true;
616.   kanna_time = System.currentTimeMillis();
617.   docchi.add(KANNA);
618.   }
619.   if (finishjikan == 36) {
620.   divflag_kanna = false;
621.   finishflg = true;
622.   countTextV.setText("分¥n析¥n中");
623.   dostop();
624.   }
625.   bureList.add(gyroValue);
626.   if (maxInt < -y_SensorValue) {
627.   maxInt = (int) -y_SensorValue;
628.   }
629.   if (minInt > -y_SensorValue) {
630.   minInt = (int) -y_SensorValue;
631.   }
632.   goukei = goukei + Math.abs(y_SensorValu

```

```

e);
608. x_kosei2 = ((x_kosei) * 10 - (x_button) *
10) / 10;
609. goukei_x = goukei_x + Math.abs(x_kosei2);
610. if (jikan == 1) {
611. unixTime = System.currentTimeMillis();
612. AccKako = Acc;
613. } else {
614. t = System.currentTimeMillis() - unixTim
e;
615. unixTime = System.currentTimeMillis();
616. Acc = -acc[1];
617. if (t > 10) {
618. kessonchi = 5;
619. } else {
620. if (kessonchi > 0) {
621. kessonchi--;
622. if (Acc <= 0.5f && Acc >= -0.5f) {
623. Spa = AccKako * t;
624. YlistA.add(Marumeru(AccKako,1));
625. } else {
626. Spa = Acc * t;
627. AccKako = Acc;
628. YlistA.add(Marumeru(Acc,1));
629. }
630. } else {
631. Spa = Acc * t;
632. YlistA.add(Marumeru(Acc,1));
633. }
634. kakoVel = Vel;
635. Vel = Vel + Spa;
636. Ylist.add(Vel);
637. }
638. }
639. saveList.add(username + "," + practiceNo
+ ","
640. + System.currentTimeMillis() + "," + acc
[0] + "," + acc[1]
641. + "," + acc[2] + "," + gyr[0] + "," + gyr
[1] + "," + gyr[2]
642. + "," + Data + ",");
643. blurList.add(gyroValue);
644. speedList.add(postSpeed);
645. koshilist.add(Float.valueOf(Data));
646. VgraphView.invalidate();
647. AgraphView.invalidate();
648. }
649. }
650. private void dostop() {
651. isRepeat = false;
652. pastY = pastYs;
653. xGraph = 20;
654. boolean isGraphStart = false;
655. boolean isGraphZero = false;
656. float max = 0;
657. int maxLim = 3;
658. boolean isMax = false;
659. float y2 = 0;
660. float yA2 = 0;
661. float henkyoku = 0;
662. int henkyokuten = 0;
663. float kyokusho = 1000000;
664. float kyokushoA = 0;
665. int kyokushoten = 0;
666. float down = 0;
667. int zeroten = 0;
668. for (int i = 0; i < Ylist.size(); i++) {
669. y2 = Ylist.get(i);
670. if (max <= y2) {
671. max = y2;
672. maxLim = 3;
673. }
674. if (!isGraphStart) {
675. zureInt++;
676. }
677. if (!isGraphStart && y2 >= 250) {
678. isGraphStart = true;
679. }
680. if (isGraphStart && max > y2) {
681. maxLim--;
682. }
683. if (maxLim == 0) {
684. isMax = true;
685. }
686. if (isMax && i > 0) {
687. if (henkyoku <= y2 - Ylist.get(i - 1)) {
688. henkyoku = y2 - Ylist.get(i - 1);
689. henkyokuten = i - 7;
690. }
691. }
692. if (isMax && kyokusho >= y2) {
693. kyokusho = y2;
694. kyokushoten = i;
695. kyokushoA = yA2;
696. }
697. if (!isGraphZero && isMax && y2 <= 0) {
698. isGraphZero = true;
699. zeroten = i;
700. }
701. }
702. down = kyokushoten - henkyokuten;
703. if (down < 0) {
704. down = 7;
705. }
706. for (int i = 0; i < Ylist.size(); i++) {
707. y2 = Ylist.get(i);
708. yA2 = YlistA.get(i);
709. if (isGraphZero) {
710. if (i >= zeroten) {
711. y2 = 0;
712. yA2 = 0;
713. }
714. } else {
715. if (i >= kyokushoten) {
716. y2 = 0;
717. yA2 = 0;
718. } else if (i >= henkyokuten) {
719. y2 = y2 - (((i - henkyokuten) / down) * ky
okusho);
720. yA2 = yA2 + (((i - henkyokuten) / down) *
kyokushoA);
721. }
722. }
723. Ylist2.add(y2);
724. YlistA2.add(yA2);
725. replayPosition = 0;
726. model_replayPosition = 0;
727. }
728. threads = null;
729. stop = System.currentTimeMillis();
730. alltime = (stop - start) / 1000;
731. if (alltime == 0) {
732. alltime = 1;
733. }
734. float speedMax = Float.parseFloat(Maximum
(Ylist2)[0]);
735. int modelSpeed = (int) Float.parseFloat(1
ogsttr);
736. speedPoint = (int) ((100 * speedMax) / mo
delSpeed);
737. if (speedPoint > 100) {
738. speedPoint = 100;
739. }
740. if (speedPoint < speedBorder1) {
741. speedRank = C;
742. speedRankStr = "☆☆☆";

```

```

743. } else if (speedPoint >= speedBorder2) {
744. speedRank = A;
745. speedRankStr = "★★★";
746. } else {
747. speedRank = B;
748. speedRankStr = "★★☆";
749. }
750. String[] bureResult = Maximum(bureList);
751. float bure = Float.parseFloat(bureResult
[0]);
752. burePoint = 0;
753. if (bure == 0f) {
754. burePoint = 100;
755. } else {
756. burePoint = (int) (Marumeru(((100 * model
Bure) / bure), 0));
757. }
758. if (burePoint > 100) {
759. burePoint = 100;
760. }
761. if (burePoint < bureBorder1) {
762. blurRank = C;
763. str_rank = "☆☆☆";
764. } else if (burePoint >= bureBorder2) {
765. blurRank = A;
766. str_rank = "★★★";
767. } else {
768. blurRank = B;
769. str_rank = "★★☆";
770. }
771. saveBure = String.valueOf(bure);
772. if (speedRank == A) {
773. if (blurRank == A) {
774. result = "ブレもなく、良いスピードで引くこと
ができています。その調子!";
775. } else if (blurRank == B) {
776. result = "良いスピードで引けていますが、少し
ブレがあります。もう少し丁寧に引いてみましょう。
";
777. } else if (blurRank == C) {
778. result = "良いスピードで引けていますが、ブレ
が大きいです。力任せに引くのではなく、丁寧さも
心掛けましょう。";
779. }
780. } else if (speedRank == B) {
781. if (blurRank == A) {
782. result = "ブレることなく丁寧に引けています。
次はそのままもう少し速く引いてみましょう。";
783. } else if (blurRank == B) {
784. result = "少しかんながブレているようです。丁
寧に、かつもう少し速く引いてみましょう。";
785. } else if (blurRank == C) {
786. result = "かんなのブレが大きいです。もっと丁
寧に、かつもう少し速く引いてみましょう。";
787. }
788. } else if (speedRank == C) {
789. if (blurRank == A) {
790. result = "丁寧に引くことができていますが、少
し遅いです。もっと速く引いてみましょう。";
791. } else if (blurRank == B) {
792. result = "引く速度が遅いです。少しブレがある
ので、丁寧さを忘れないようにしつつ、もっと速く
引いてみましょう。";
793. } else if (blurRank == C) {
794. result = "ブレが大きく、引く速度も遅いです。
丁寧さを忘れないようにしつつ、もっと速く引いて
みましょう。";
795. }
796. }
797. if (docchi.size() == 2) {
798. docchif = (kanna_time - koshi_time) / 100
0f;
799. if (docchif > 0) {
800. docchif = (kanna_time - koshi_time) / 100
0f;
801. timing = "腰から全身でしっかり引くことができ
ています! ¥n ランク: ★★★";
802. saveKoshi = "KOSHI->TE";
803. koshiPoint = 100;
804. } else {
805. timing = "手元が先に動いています。腰から全身
を使って引きましょう! ¥n ランク: ★☆☆";
806. saveKoshi = "TE->KOSHI";
807. koshiPoint = 70;
808. }
809. saveTime = String.valueOf(docchif);
810. } else {
811. timing = "上半身だけで引いています。腰から全
身で引きましょう! ¥n ランク: ★☆☆";
812. saveKoshi = "TE only";
813. saveTime = "-";
814. koshiPoint = 30;
815. }
816. ViewStr = "¥n【引きの速さとブレ】" + "¥n師匠
を100点とすると..." + "¥n あなたの引きの速さ
は" + speedPoint
+ "点です。" + "¥n ブレの少なさは" + burePo
int + "点です。" + "¥n 速さランク:"
+ speedRankStr + "¥n ブレランク:" + str_r
ank + "¥n" + result
+ "¥n¥n【全身の使い方】" + "¥n" + timing;
820. textV.setTextColor(Color.WHITE);
821. textV.setText("計測完了");
822. textV.setText(ViewStr);
823. textV.setTextSize(17);
824. textV.setPadding(0, 10, 0, 0);
825. textV.setGravity(Gravity.LEFT);
826. startFlag = false;
827. mp = MediaPlayer.create(this, R.raw.resul
t);
828. try {
829. mp.prepare();
830. } catch (IllegalStateException e) {
831. e.printStackTrace();
832. } catch (IOException e) {
833. e.printStackTrace();
834. }
835. mp.start();
836. saveText = "";
837. for (int i = 0; i < saveList.size(); i++)
{
838. saveText += "¥n" + saveList.get(i);
839. if (i == saveList.size() - 1) {
840. saveText += saveKoshi + "," + saveTime;
841. }
842. }
843. saveHosei = "";
844. for (int i = 0; i < Ylist2.size(); i++) {
845. saveHosei += "¥n" + username + "," + pract
iceNo + "," + Ylist2.get(i) + "," + YlistA
2.get(i) + "," + bureList.get(i);
846. }
847. saveHosei += "," + speedPoint + "," + spee
dRank + "," + burePoint + "," + blurRank +
"," + saveKoshi;
848. try {
849. BufferedWriter bw = new BufferedWriter(new
OutputStreamWriter(
new FileOutputStream(file, true), "UTF-8
"));
851. bw.write(saveText);
852. bw.close();
853. BufferedWriter bwHosei = new BufferedWrit
er(new OutputStreamWriter(
new FileOutputStream(fileHosei, true), "U
TF-8"));

```

```

855. bwHosei.write(saveHosei);
856. bwHosei.close();
857. Toast.makeText(this, "Saved", Toast.LENGTH_SHORT).show();
858. } catch (UnsupportedEncodingException k) {
859.     k.printStackTrace();
860.     Toast.makeText(this, "保存失敗", Toast.LENGTH_SHORT).show();
861. } catch (FileNotFoundException k) {
862.     k.printStackTrace();
863.     Toast.makeText(this, "保存失敗", Toast.LENGTH_SHORT).show();
864. } catch (IOException k) {
865.     k.printStackTrace();
866.     Toast.makeText(this, "保存失敗", Toast.LENGTH_SHORT).show();
867. }
868. time.setToNow();
869. String now = String.valueOf(time.hour) +
870.     String.valueOf(time.minute)
871.     + String.valueOf(time.second);
872. String filePath = Environment.getExternalStorageDirectory().getPath()
873.     + "/" + getPackageName() + "/two/" + username + "_"
874.     /* + System.currentTimeMillis() + "_" */ +
875.     (cld.get(cld.YEAR))
876.     + "_" + cld.get(cld.MONTH) + "_" + cld.get(cld.DAY_OF_MONTH)
877.     + "_" + cld.get(cld.HOUR_OF_DAY) + "_" +
878.     cld.get(cld.MINUTE)
879.     + "_" + cld.get(cld.SECOND) + "_" + practiceNo + ".png";
880. File file = new File(filePath);
881. file.getParentFile().mkdirs();
882. FileOutputStream fos;
883. practiceNo++;
884. try {
885.     fos = new FileOutputStream(file, true);
886.     bmp.compress(CompressFormat.PNG, 10, fos);
887.     fos.flush();
888.     fos.close();
889. } catch (Exception e) {
890. }
891. mHandler.removeCallbacks(looper);
892. postToDB();
893. LayoutInflater inflater = LayoutInflater.from(Kanna2Activity.this);
894. resultView = inflater.inflate(R.layout.result_view, null);
895. TabHost host = (TabHost) resultView.findViewById(R.id.tabhost);
896. host.setup();
897. TabSpec tabComment = host.newTabSpec("Comment");
898. tabComment.setIndicator("コメント");
899. tabComment.setContent(R.id.resultComment);
900. host.addTab(tabComment);
901. TabSpec tabGraph = host.newTabSpec("Graph");
902. tabGraph.setIndicator("グラフ");
903. tabGraph.setContent(R.id.resultGraph);
904. host.addTab(tabGraph);
905. TabSpec tabReplay = host.newTabSpec("Replay");
906. tabReplay.setIndicator("リプレイ");
907. tabReplay.setContent(R.id.resultReplay);
908. host.addTab(tabReplay);
909. replayView = new ReplayView(Kanna2Activity.this);
910. replayBtn2 = new Button(Kanna2Activity.this);
911. replayBtn2.setText("リプレイ");
912. replayBtn2.setLayoutParams(new LayoutParams(
913.     LayoutParams.WRAP_CONTENT, LayoutParams.WRAP_CONTENT));
914. replayBtn2.setOnClickListener(new OnClickListener() {
915.     public void onClick(View v) {
916.         replayPosition = 0;
917.         model_replayPosition = 0;
918.         replayer = new Runnable() {
919.             public void run() {
920.                 mHandler.post(new Runnable() {
921.                     public void run() {
922.                         replayView.invalidate();
923.                     }
924.                 });
925.             }
926.         };
927.         try {
928.             Thread.sleep(500);
929.         } catch (InterruptedException e) {
930.         }
931.         for (int i = zureInt; i < Ylist2.size()
932.             || modelZureInt + i - zureInt < model_Ylist.size(); i++) {
933.             if (i < Ylist2.size()) {
934.                 replayPosition += Ylist2.get(i);
935.             } else {
936.                 replayPosition += 0;
937.             }
938.             if (modelZureInt + i - zureInt < model_Ylist.size()) {
939.                 model_replayPosition += model_Ylist.get(modelZureInt + i - zureInt);
940.             } else {
941.                 model_replayPosition += 0;
942.             }
943.             try {
944.                 Thread.sleep(REPEAT_INTERVAL * 5);
945.             } catch (InterruptedException e) {
946.             }
947.             mHandler.post(new Runnable() {
948.                 public void run() {
949.                     replayView.invalidate();
950.                 }
951.             });
952.         }
953.         replay = new Thread(replayer);
954.         replay.start();
955.     }
956. });
957. replayDlg = new AlertDialog.Builder(Kanna2Activity.this);
958. replayDlg.setView(resultView);
959. VgraphView = new VgraphView(Kanna2Activity.this);
960. AgraphView = new AgraphView(Kanna2Activity.this);
961. TextView rt = (TextView) resultView.findViewById(R.id.resultText);
962. LinearLayout g = (LinearLayout) resultView.findViewById(R.id.resultGraph);
963. LinearLayout r = (LinearLayout) resultView.findViewById(R.id.resultReplay);
964. rt.setText(ViewStr);
965. rt.setTextSize(17);
966. rt.setTextColor(Color.WHITE);
967. LinearLayout.LayoutParams lp = new LinearLayout.LayoutParams(
968.     LayoutParams.FILL_PARENT, LayoutParams.WR

```

```

    AP_CONTENT);
966. lp.weight = 1;
967. g.addView(VgraphView, lp);
968. g.addView(AgraphView, lp);
969. r.setGravity(Gravity.CENTER_HORIZONTAL);
970. r.addView(replayBtn2);
971. r.addView(replayView);
972. replayDlg.setCancelable(false);
973. replayDlg.setNegativeButton("次へ",
974. new DialogInterface.OnClickListener() {
975. public void onClick(DialogInterface dialo
976. g, int id) {
977. webView();
978. }
979. });
980. replayDlg.show();
981. private class VgraphView extends View {
982. public VgraphView(Context context) {
983. super(context);
984. setFocusable(true);
985. }
986. protected void onSizeChanged(int w, int h,
987. int oldw, int oldh) {
988. super.onSizeChanged(w, h, oldw, oldh);
989. wide = w;
990. hight = h;
991. pastYs = h / 2;
992. pastY = (hight / 4) * 3;
993. pastYA = (hight / 4) * 2;
994. if (bmp == null) {
995. bmp_model = Bitmap.createBitmap(w, h, Bit
996. map.Config.ARGB_4444);
997. bmp_model = BitmapFactory.decodeFile(Envi
998. ronment.getExternalStorageDirectory().get
999. Path()
1000. + "/"
1001. + getPackageName() + "/two/" + modelName
1002. + ".png");
1003. bmpCanvas = new Canvas(bmp);
1004. bmpCanvas.drawColor(Color.argb(0, 200, 25
1005. 5, 255));
1006. path.moveTo(20, (hight / 4) * 3);
1007. pathA.moveTo(20, (hight / 4) * 2);
1008. span = (wide-40)/360f;
1009. for (int i = 0; i < Ylist2.size(); i++) {
1010. mYOffset = MoveAverage(i)[0];
1011. mYOffsetA = MoveAverage(i)[1];
1012. int y = (int) (mYOffset);
1013. int yA = (int) (mYOffsetA);
1014. path.quadTo(xGraph, pastY, (xGraph + xGra
1015. ph + span) / 2,
1016. (pastY + y) / 2);
1017. pathA.quadTo(xGraph, pastYA, (xGraph + xG
1018. raph + span) / 2,
1019. (pastYA + yA) / 2);
1020. xGraph = xGraph + span;
1021. pastY = y;
1022. pastYA = yA;
1023. }
1024. bmpCanvas.drawLine(20, hight / 2, wide -
1025. 20, hight / 2, paintLine2);
1026. bmpCanvas.drawLine(20, (hight / 4) * 1, w
1027. ide - 20, (hight / 4) * 1,
1028. paintLine2);
1029. bmpCanvas.drawLine(20, (hight / 4) * 3, w
1030. ide - 20, (hight / 4) * 3,
1031. paintLine);
1032. bmpCanvas.drawPath(path, paintSensor);
1033. bmpCanvas.drawCircle(20 + (koshi_time_int
-30)*span, (hight / 4) * 3, 10, paintSens
or);
1034. bmpCanvas.drawLine(5, hight, wide - 5, hi
1035. ght, paintBorder);
1036. VgraphView.invalidate();
1037. }
1038. @Override
1039. protected void onDraw(Canvas canvas) {
1040. canvas.drawBitmap(bmp, 0, 0, null);
1041. if (isRepeat) {
1042. bmpCanvas.drawLine(20, hight / 2, wide -
1043. 20, hight / 2,
1044. paintLine2);
1045. bmpCanvas.drawLine(20, (hight / 4) * 1, w
1046. ide - 20,
1047. (hight / 4) * 1, paintLine2);
1048. bmpCanvas.drawLine(20, (hight / 4) * 3, w
1049. ide - 20,
1050. (hight / 4) * 3, paintLine);
1051. if (startFlag == true) {
1052. if (koshiPlot) {
1053. koshiPlot = false;
1054. }
1055. bmpCanvas.drawLine(5, hight, wide - 5, hi
1056. ght, paintBorder);
1057. if (isKannaMove) {
1058. isKannaMove = false;
1059. }
1060. } else {
1061. canvas.drawBitmap(bmp_model, (zureInt-mod
1062. elZureInt)*span, 0, null);
1063. paintText.setColor(Color.WHITE);
1064. paintText.setTextSize(35);
1065. canvas.drawText("速度", 10, 55, paintTex
1066. t);
1067. paintText.setColor(Color.CYAN);
1068. paintText.setTextSize(25);
1069. canvas.drawText("あなた", 150, 55, paintTe
1070. xt);
1071. paintText.setColor(Color.YELLOW);
1072. paintText.setTextSize(25);
1073. canvas.drawText("師匠", 250, 55, paintTex
1074. t);
1075. }
1076. }
1077. private class AgraphView extends View {
1078. public AgraphView(Context context) {
1079. super(context);
1080. setFocusable(true);
1081. }
1082. protected void onSizeChanged(int w, int h,
1083. int oldw, int oldh) {
1084. super.onSizeChanged(w, h, oldw, oldh);
1085. wide = w;
1086. hight = h;
1087. pastYs = h / 2;
1088. pastY = h / 2;
1089. if (bmpA == null) {
1090. bmpA = Bitmap.createBitmap(w, h, Bitmap.C
1091. onfig.ARGB_4444);
1092. bmp_modelA = Bitmap.createBitmap(w, h, Bi
1093. tmap.Config.ARGB_4444);
1094. bmp_modelA =
1095. BitmapFactory.decodeFile(Environment.getE
1096. xternalStorageDirectory().getPath()
1097. + "/" + getPackageName() + "/two/" + mode
1098. lName + "A.png");
1099. }
1100. bmpCanvasA = new Canvas(bmpA);
1101. bmpCanvasA.drawColor(Color.argb(0, 200, 2
1102. 55, 255));

```

```

1081. bmpCanvasA.drawLine(20, hight / 2, wide -
    20, hight / 2, paintLine);
1082. bmpCanvasA.drawLine(20, (hight / 4) * 1,
    wide - 20,
1083. (hight / 4) * 1, paintLine2);
1084. bmpCanvasA.drawLine(20, (hight / 4) * 3,
    wide - 20,
1085. (hight / 4) * 3, paintLine2);
1086. bmpCanvasA.drawPath(pathA, paintSensor);
1087. bmpCanvasA.drawCircle(20 + (koshi_time_in
    t-30)*span, hight / 2, 10, paintSensor);
1088. bmpCanvasA.drawLine(5, 0, wide - 5, 0, pa
    intBorder);
1089. AgraphView.invalidate();
1090. }
1091. @Override
1092. protected void onDraw(Canvas canvas) {
1093. canvas.drawBitmap(bmpA, 0, 0, null);
1094. if (isRepeat) {
1095. bmpCanvasA.drawLine(5, 0, wide - 5, 0, pa
    intBorder);
1096. bmpCanvasA.drawLine(20, hight / 2, wide -
    20, hight / 2,
1097. paintLine);
1098. bmpCanvasA.drawLine(20, (hight / 4) * 1,
    wide - 20,
1099. (hight / 4) * 1, paintLine2);
1100. bmpCanvasA.drawLine(20, (hight / 4) * 3,
    wide - 20,
1101. (hight / 4) * 3, paintLine2);
1102. } else {
1103. canvas.drawBitmap(bmp_modelA, (zureInt-mo
    delZureInt)*span, 0, null);
1104. paintText.setTextSize(35);
1105. paintText.setColor(Color.WHITE);
1106. canvas.drawText("加速度", 10, 55, paintTex
    t);
1107. }
1108. }
1109. }
1110. @Override
1111. public void onResume() {
1112. super.onResume();
1113. Sensor sensor = sensorManager
1114. .getDefaultSensor(Sensor.TYPE_LINEAR_ACCE
    LERATION);
1115. Sensor sensor2 = sensorManager.getDefault
    Sensor(Sensor.TYPE_GYROSCOPE);
1116. sensorManager.registerListener(this, sens
    or,
1117. SensorManager.SENSOR_DELAY_FASTEST);
1118. sensorManager.registerListener(this, sens
    or2,
1119. SensorManager.SENSOR_DELAY_FASTEST);
1120. }
1121. @Override
1122. public void onPause() {
1123. super.onPause();
1124. sensorManager.unregisterListener(this);
1125. }
1126. public void onSensorChanged(SensorEvent e
    vent) {
1127. switch (event.sensor.getType()) {
1128. case Sensor.TYPE_LINEAR_ACCELERATION:
1129. acc = event.values;
1130. y_sensor = event.values[1];
1131. x_sensor = event.values[0];
1132. postSpeed = SC.setNowSensorValue(-y_sens
    or);
1133. break;
1134. case Sensor.TYPE_GYROSCOPE:
1135. gyr = event.values;
1136. gyro_value[0] = Math.abs(Marumeru(event.v
    alues[0], 2));
1137. gyro_value[1] = Math.abs(Marumeru(event.v
    alues[1], 2));
1138. gyro_value[2] = Math.abs(Marumeru(event.v
    alues[2], 2));
1139. gyroValue = Marumeru(event.values[1], 2)
    + Marumeru(event.values[2], 2);
1140. + Marumeru(event.values[2], 2);
1141. break;
1142. }
1143. }
1144. public void kezuroukai() {
1145. e.putString("username", "kezuroukai");
1146. e.putString("userid", "kezuroukai");
1147. e.commit();
1148. username = "kezuroukai";
1149. cld = Calendar.getInstance();
1150. filename = Environment.getExternalStorage
    Directory().getPath() + "/"
1151. + getPackageName() + "/two/" + username +
    " "
1152. + (cld.get(cld.YEAR))
1153. + "_" + cld.get(cld.MONTH) + "_" + cld.ge
    t(cld.DAY_OF_MONTH)
1154. + "_" + cld.get(cld.HOUR_OF_DAY) + "_" +
    cld.get(cld.MINUTE)
1155. + "_" + cld.get(cld.SECOND) + "_nama.csv
    ";
1156. file = new File(filename);
1157. filenameHosei = Environment.getExternalSt
    orageDirectory().getPath() + "/"
1158. + getPackageName() + "/two/" + username +
    " "
1159. /* + System.currentTimeMillis() + "_" */+
    (cld.get(cld.YEAR))
1160. + "_" + cld.get(cld.MONTH) + "_" + cld.ge
    t(cld.DAY_OF_MONTH)
1161. + "_" + cld.get(cld.HOUR_OF_DAY) + "_" +
    cld.get(cld.MINUTE)
1162. + "_" + cld.get(cld.SECOND) + "_hosei.csv
    ";
1163. fileHosei = new File(filenameHosei);
1164. file.getParentFile().mkdirs();
1165. saveText = "id,times,unix_time,acc_x,acc_
    y,acc_z,gyr_x,gyr_y,gyr_z,koshi_acc,body,
    body_gap";
1166. saveHosei = "Acc,Speed,Koshi,speedPoint,s
    peedRank,burePoint,blurRank,saveKoshi";
1167. try {
1168. BufferedWriter bw = new BufferedWriter(ne
    w OutputStreamWriter(
1169. new FileOutputStream(file, true), "UTF-8
    ");
1170. bw.write(saveText);
1171. bw.close();
1172. BufferedWriter bwHosei = new BufferedWrit
    er(new OutputStreamWriter(
1173. new FileOutputStream(fileHosei, true), "U
    TF-8"));
1174. bwHosei.write(saveHosei);
1175. bwHosei.close();
1176. } catch (UnsupportedEncodingException k)
    {
1177. k.printStackTrace();
1178. } catch (FileNotFoundException k) {
1179. k.printStackTrace();
1180. } catch (IOException k) {
1181. k.printStackTrace();
1182. }
1183. mDevices = new Vector<BluetoothDevice>();
1184. mAdapter = BluetoothAdapter.getDefaultAda
    pter();
1185. Set<BluetoothDevice> devices = mAdapter.g
    etBondedDevices();

```

```

1186. for (BluetoothDevice device : devices) {
1187.     mDevices.add(device);
1188. }
1189. serverThread = new ServerThread();
1190. serverThread.start();
1191. }
1192. }
1193. public void getuser() {
1194.     String s = host + "studentsData.txt";
1195.     HttpGet method = new HttpGet(s.trim().toString());
1196.     String src = "";
1197.     try {
1198.         DefaultHttpClient client = new DefaultHttpClient();
1199.         HttpResponse response = client.execute(method);
1200.         src = EntityUtils.toString(response.getEntity(), "UTF-8");
1201.         String[] strAry = src.split("\n");
1202.         ArrayList<String> arrayName = new ArrayList<String>();
1203.         ArrayList<String> arrayGakuban = new ArrayList<String>();
1204.         ArrayList<String> arrayKeyId = new ArrayList<String>();
1205.         for (int i = 0; i < strAry.length; i++) {
1206.             String[] strAryPerson = strAry[i].split("\n");
1207.             if (Integer.parseInt(strAryPerson[1]) == userClassNum) {
1208.                 arrayGakuban.add(strAryPerson[2]);
1209.                 arrayName.add(strAryPerson[3]);
1210.                 arrayKeyId.add(strAryPerson[5]);
1211.             }
1212.         }
1213.     } catch (Exception e) {
1214.     } finally {
1215.     }
1216.     listView();
1217. }
1218. public void listView() {
1219.     ListView LV = new ListView(this);
1220.     adapter = new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, arrayGakuban);
1221.     LV.setAdapter(adapter);
1222.     LV.setOnItemClickListener(new OnItemClickListener() {
1223.         public void onItemClick(AdapterView<?> items, View view,
1224.             int position, long id) {
1225.             e.putString("username", arrayGakuban.get(position).toString());
1226.             e.putString("userid", arrayKeyId.get(position).toString().trim());
1227.             userId = Integer.valueOf(arrayKeyId.get(position).toString().trim());
1228.             e.commit();
1229.             ID = arrayKeyId.get(position).toString().trim();
1230.             username = arrayGakuban.get(position).toString();
1231.             if (username == null || username.equals("")) {
1232.                 username = "foo";
1233.             }
1234.             cld = Calendar.getInstance();
1235.             filename = Environment.getExternalStorageDirectory().getPath() + "/"
1236.                 + getPackageName() + "/two/" + username +
1237.                 + cld.get(cld.YEAR)
1238.                 + "_" + cld.get(cld.MONTH) + "_" + cld.get(cld.DAY_OF_MONTH)
1239.                 + "_" + cld.get(cld.HOUR_OF_DAY) + "_" +
1240.                 cld.get(cld.MINUTE)
1241.                 + "_" + cld.get(cld.SECOND) + "_nama.csv";
1242.             file = new File(filename);
1243.             filenameHosei = Environment.getExternalStorageDirectory().getPath() + "/"
1244.                 + getPackageName() + "/two/" + username +
1245.                 + cld.get(cld.YEAR)
1246.                 + "_" + cld.get(cld.MONTH) + "_" + cld.get(cld.DAY_OF_MONTH)
1247.                 + "_" + cld.get(cld.HOUR_OF_DAY) + "_" +
1248.                 cld.get(cld.MINUTE)
1249.                 + "_" + cld.get(cld.SECOND) + "_hosei.csv";
1250.             fileHosei = new File(filenameHosei);
1251.             file.getParentFile().mkdirs();
1252.             saveText = "id,times,unix_time,acc_x,acc_y,acc_z,gyr_x,gyr_y,gyr_z,koshi_acc,body,
1253.                 body_gap";
1254.             saveHosei = "Acc,Speed,Koshi,speedPoint,speedRank,burePoint,blurRank,saveKoshi";
1255.             try {
1256.                 BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(
1257.                     new FileOutputStream(file, true), "UTF-8"));
1258.                 bw.write(saveText);
1259.                 bw.close();
1260.                 BufferedWriter bwHosei = new BufferedWriter(new OutputStreamWriter(
1261.                     new FileOutputStream(fileHosei, true), "UTF-8"));
1262.                 bwHosei.write(saveHosei);
1263.                 bwHosei.close();
1264.             } catch (UnsupportedEncodingException k) {
1265.             }
1266.             k.printStackTrace();
1267.             } catch (FileNotFoundException k) {
1268.             }
1269.             k.printStackTrace();
1270.             } catch (IOException k) {
1271.             }
1272.             k.printStackTrace();
1273.             countBtn.setEnabled(true);
1274.             changeUser.setEnabled(true);
1275.             userTextV.setText("学習者:" + arrayName.get(position).toString());
1276.             dialog2.dismiss();
1277.         }
1278.     });
1279.     dialog = new AlertDialog.Builder(this);
1280.     dialog2 = dialog.create();
1281.     dialog2.setCanceledOnTouchOutside(false);
1282.     dialog2.setTitle("出席番号を選択してください");
1283.     dialog2.setView(LV);
1284.     dialog2.show();
1285. }
1286. public String postToDB() {
1287.     String speed = "";
1288.     String acc = "";
1289.     String blur = "";
1290.     String koshi = "";
1291.     for (int i = 0; i < Ylist2.size(); i++) {
1292.         speed += Math.round(Ylist2.get(i));
1293.         if (i != Ylist2.size() - 1) {
1294.             speed += ",";
1295.         }
1296.     }
1297.     for (int i = 0; i < YlistA2.size(); i++) {
1298.         acc += Math.round(YlistA2.get(i));
1299.         if (i != YlistA2.size() - 1) {
1300.             acc += ",";
1301.         }
1302.     }

```

```

1297. }
1298. }
1299. for (int i = 0; i < blurList.size(); i++)
1300. {
1301. blur += Math.round(blurList.get(i));
1302. if (i != blurList.size() - 1) {
1303. blur += ",";
1304. }
1305. }
1306. for (int i = 0; i < koshiList.size(); i++)
1307. {
1308. koshi += Math.round(koshiList.get(i));
1309. if (i != koshiList.size() - 1) {
1310. koshi += ",";
1311. }
1312. String s = host + "plain/senddata.php?ID="
1313. + ID
1314. + "&speed=" + speed
1315. + "&acc=" + acc
1316. + "&gyro=" + blur
1317. + "&koshi=" + koshi
1318. + "&gap=" + zureInt
1319. + "&accPoint=" + speedPoint
1320. + "&gyroPoint=" + burePoint
1321. + "&koshiPoint=" + koshiPoint
1322. + "&totalPoint=" + (speedPoint + burePoint
1323. + koshiPoint)/3;
1324. HttpGet method = new HttpGet(s.toString
1325. ());
1326. try {
1327. DefaultHttpClient client = new DefaultHtt
1328. pClient();
1329. HttpResponse response = client.execute(me
1330. thod);
1331. EntityUtils.toString(response.getEntity(),
1332. "UTF-8");
1333. return EntityUtils.toString(response.getE
1334. ntity(), "UTF-8");
1335. } catch (Exception e) {
1336. return null;
1337. }
1338. }
1339. public void onAccuracyChanged(Sensor sens
1340. or, int accuracy) {
1341. }
1342. }
1343. private class ServerThread extends Thread
1344. {
1345. private final BluetoothServerSocket mmSer
1346. verSocket;
1347. public ServerThread() {
1348. BluetoothServerSocket tmp = null;
1349. try {
1350. tmp = mAdapter.listenUsingRfcommWithServi
1351. ceRecord(NAME, MY_UUID);
1352. } catch (IOException e) {
1353. }
1354. mmServerSocket = tmp;
1355. }
1356. @Override
1357. public void run() {
1358. while (true) {
1359. try {
1360. socket[0] = mmServerSocket.accept();
1361. } catch (IOException e) {
1362. }
1363. if (socket[0] != null) {
1364. manageConnectedSocket(socket[0]);
1365. Message msg = new Message();
1366. msg.obj = "";
1367. toastHandler.sendMessage(msg);
1368. connection[0].start();
1369. }
1370. }
1371. }
1372. break;
1373. }
1374. }
1375. @SuppressWarnings("unused")
1376. public void cancel() {
1377. try {
1378. mmServerSocket.close();
1379. } catch (IOException e) {
1380. }
1381. }
1382. }
1383. private class ConnectedThread extends Thr
1384. ead {
1385. private final BluetoothSocket mmSocket;
1386. private final InputStream mmInStream;
1387. private final OutputStream mmOutStream;
1388. String[] strAry = new String[5];
1389. public ConnectedThread(BluetoothSocket so
1390. cket) {
1391. mmSocket = socket;
1392. InputStream tmpIn = null;
1393. OutputStream tmpOut = null;
1394. try {
1395. tmpIn = socket.getInputStream();
1396. tmpOut = socket.getOutputStream();
1397. } catch (IOException e) {
1398. }
1399. mmInStream = tmpIn;
1400. mmOutStream = tmpOut;
1401. }
1402. @Override
1403. public void run() {
1404. byte[] buffer = new byte[1024];
1405. @SuppressWarnings("unused")
1406. int bytes;
1407. while (true) {
1408. try {
1409. bytes = mmInStream.read(buffer);
1410. String readMsg = new String(buffer, "UTF-
1411. 8");
1412. Message msg = new Message();
1413. msg.obj = readMsg;
1414. strAry = readMsg.split("ita00");
1415. Data = strAry[0];
1416. } catch (IOException e) {
1417. }
1418. break;
1419. }
1420. }
1421. @SuppressWarnings("unused")
1422. public void write(byte[] bytes) {
1423. try {
1424. mmOutStream.write(bytes);
1425. } catch (IOException e) {
1426. }
1427. }
1428. @SuppressWarnings("unused")
1429. public void cancel() {
1430. try {
1431. mmSocket.close();
1432. } catch (IOException e) {
1433. }
1434. }
1435. public void manageConnectedSocket(Bluetoo
1436. thSocket socket) {
1437. connection[0] = new ConnectedThread(socke
1438. t);
1439. }
1440. final Handler toastHandler = new Handler
1441. () {
1442. public void handleMessage(Message msg) {
1443. Toast.makeText(Kanna2Activity.this, msg.o

```

```

    bj + "接続完了",
1424. Toast.LENGTH_SHORT).show();
1425. classView();
1426. }
1427. };
1428. public boolean onKeyDown(int keyCode, KeyEvent event) {
1429.     if (keyCode == KeyEvent.KEYCODE_BACK) {
1430.         System.exit(0);
1431.     } else if (keyCode == KeyEvent.KEYCODE_MENU) {
1432.         openOptionsMenu();
1433.     }
1434.     return true;
1435. }
1436. public float Marumeru(float f, int i) {
1437.     BigDecimal xbi = new BigDecimal(String.valueOf(f));
1438.     float result = xbi.setScale(i, BigDecimal.ROUND_HALF_UP).floatValue();
1439.     return result;
1440. }
1441. public String[] Maximum(ArrayList<Float> list) {
1442.     String[] result = new String[3];
1443.     float f = 0f;
1444.     int index = 0;
1445.     String when = "";
1446.     for (int i = 0; i < list.size(); i++) {
1447.         if (f < list.get(i)) {
1448.             f = list.get(i);
1449.             index = i;
1450.         }
1451.     }
1452.     if (index < list.size() / 3f) {
1453.         when = "引き始め";
1454.     } else if (index >= list.size() / 1.5f) {
1455.         when = "引き終わり";
1456.     } else {
1457.         when = "引いている途中";
1458.     }
1459.     result[0] = String.valueOf(f);
1460.     result[1] = String.valueOf(index);
1461.     result[2] = when;
1462.     return result;
1463. }
1464. public float[] MoveAverage(int i) {
1465.     float[] result = new float[2];
1466.     float Vsum = 0;
1467.     float Asum = 0;
1468.     if (i > 10 && i < Ylist2.size() - 10) {
1469.         for (int j = -5; j < 5; j++) {
1470.             Vsum += Ylist2.get(i + j);
1471.         }
1472.         for (int j = -10; j < 10; j++) {
1473.             Asum += YlistA2.get(i + j);
1474.         }
1475.         result[0] = (hight / 4) * 3 + (-Vsum / 10) / 10;
1476.         result[1] = hight / 2 + (-Asum / 20) * 7;
1477.     } else {
1478.         result[0] = (hight / 4) * 3 + (-Ylist2.get(i)) / 10;
1479.         result[1] = hight / 2 + (-YlistA2.get(i)) * 7;
1480.     }
1481.     return result;
1482. }
1483. public class ReplayView extends View {
1484.     public ReplayView(Context context) {
1485.         super(context);
1486.     }
1487.     protected void onSizeChanged(int w, int h,
        int oldw, int oldh) {
1488.         super.onSizeChanged(w, h, oldw, oldh);
1489.         replayHeight = h;
1490.         replayWidth = w;
1491.         replayRight = 1 * (replayWidth / 4);
1492.         replayCenter = 2 * (replayWidth / 4);
1493.         replayLeft = 3 * (replayWidth / 4);
1494.         replayRatio = (replayHeight - 50) / replayLength;
1495.     }
1496.     @Override
1497.     protected void onDraw(Canvas canvas) {
1498.         canvas.drawColor(Color.WHITE);
1499.         canvas.drawBitmap(kanna, replayRight, replayPosition * replayRatio, null);
1500.         canvas.drawBitmap(kanna, replayLeft, mode1_replayPosition * replayRatio, null);
1501.         paintText.setTextSize(35);
1502.         paintText.setColor(Color.BLACK);
1503.         canvas.drawText("あなた", replayRight, 100, paintText);
1504.         canvas.drawText("師匠", replayLeft, 100, paintText);
1505.     }
1506. }
1507. }
1508. public void webView() {
1509.     final WebView webView = new WebView(this);
1510.     webView.getSettings().setJavaScriptEnabled(true);
1511.     webView.loadUrl(host + "plain/growthRecordMobile.php?ID="
1512.         + userId);
1513.     AlertDialog.Builder dlg = new AlertDialog.Builder(this);
1514.     AlertDialog.Builder dialog2 = new AlertDialog.Builder(this);
1515.     dialog2.setTitle("成績表示");
1516.     dialog2.setView(webView);
1517.     dialog2.setPositiveButton("もう一度",
1518.         new DialogInterface.OnClickListener() {
1519.             public void onClick(DialogInterface dialog, int id) {
1520.                 intCount = 4;
1521.                 jikan = 0;
1522.                 koshi_time_int = 0;
1523.                 zureInt = 1;
1524.                 pastY = pastYs;
1525.                 pastYA = pastYs;
1526.                 xGraph = 20;
1527.                 countBtn.setVisibility(View.VISIBLE);
1528.                 changeUser.setVisibility(View.VISIBLE);
1529.                 Paint p = new Paint();
1530.                 p.setColor(Color.WHITE);
1531.                 if (bmpCanvas != null && bmpCanvasA != null) {
1532.                     bmpCanvas.drawColor(Color.TRANSPARENT, PorterDuff.Mode.CLEAR);
1533.                     bmpCanvasA.drawColor(Color.TRANSPARENT, PorterDuff.Mode.CLEAR);
1534.                 }
1535.                 countTextV.setText("↓");
1536.                 countTextV.setTextSize(250);
1537.                 divflag_koshi = false;
1538.                 divflag_kanna = false;
1539.                 currentlyInt_kanna = new int[10];
1540.                 currently_koshi = new float[10];
1541.                 docchi.clear();
1542.                 bureList.clear();
1543.                 speedList.clear();
1544.                 blurList.clear();
1545.                 koshiList.clear();

```

```

1546. saveList.clear();
1547. unixTime = 0;
1548. t = 0;
1549. y_hosei = 0;
1550. z_hosei = 0;
1551. Acc = 0;
1552. AccKako = 0;
1553. Spa = 0;
1554. Vel = 0;
1555. kakoVel = 0;
1556. kessonchi = 0;
1557. graphDownInt = graphDownReset;
1558. path = new Path();
1559. pathA = new Path();
1560. XYnum = 0;
1561. Ylist.clear();
1562. Ylist2.clear();
1563. YlistA.clear();
1564. YlistA2.clear();
1565. }
1566. });
1567. dialog2.setNegativeButton("学習者の変更",
1568. new DialogInterface.OnClickListener() {
1569. public void onClick(DialogInterface dialo
g, int id) {
1570. intCount = 4;
1571. jikan = 0;
1572. koshi_time_int = 0;
1573. zureInt = 1;
1574. pastY = pastYs;
1575. pastYA = pastYs;
1576. xGraph = 20;
1577. countBtn.setVisibility(View.VISIBLE);
1578. changeUser.setVisibility(View.VISIBLE);
1579. Paint p = new Paint();
1580. p.setColor(Color.WHITE);
1581. if (bmpCanvas != null && bmpCanvasA != nu
ll) {
1582. bmpCanvas.drawColor(Color.TRANSPARENT, Po
rterDuff.Mode.CLEAR);
1583. bmpCanvasA.drawColor(Color.TRANSPARENT, P
orterDuff.Mode.CLEAR);
1584. }
1585. countTextV.setText("↓");
1586. countTextV.setTextSize(250);
1587. divflag_koshi = false;
1588. divflag_kanna = false;
1589. currentlyInt_kanna = new int[10];
1590. currently_koshi = new float[10];
1591. docchi.clear();
1592. bureList.clear();
1593. speedList.clear();
1594. blurList.clear();
1595. koshiList.clear();
1596. saveList.clear();
1597. unixTime = 0;
1598. t = 0;
1599. y_hosei = 0;
1600. z_hosei = 0;
1601. Acc = 0;
1602. AccKako = 0;
1603. Spa = 0;
1604. Vel = 0;
1605. kakoVel = 0;
1606. kessonchi = 0;
1607. graphDownInt = graphDownReset;
1608. path = new Path();
1609. pathA = new Path();
1610. XYnum = 0;
1611. Ylist.clear();
1612. Ylist2.clear();
1613. YlistA.clear();
1614. YlistA2.clear();
1615. getuser();
1616. }
1617. });
1618. dialog2.show();
1619. }
1620. }

```

2-2 学習者用アプリケーション（Moodle との連携時、かなな端末のメインのプログラ

ム）

```

1. package <packageName>;
2. import android.app.Activity;
3. import android.app.AlertDialog;
4. import android.app.ProgressDialog;
5. import android.bluetooth.BluetoothAdapt
er;
6. import android.bluetooth.BluetoothDevic
e;
7. import android.bluetooth.BluetoothServe
rSocket;
8. import android.bluetooth.BluetoothSocke
t;
9. import android.content.Context;
10. import android.content.DialogInterface;
11. import android.content.Intent;
12. import android.content.SharedPreferences;
13. import android.content.SharedPreferences
s.Editor;
14. import android.graphics.Bitmap;
15. import android.graphics.BitmapFactory;
16. import android.graphics.Canvas;
17. import android.graphics.Color;
18. import android.graphics.Paint;
19. import android.graphics.Path;
20. import android.graphics.PorterDuff;
21. import android.hardware.Sensor;
22. import android.hardware.SensorEvent;
23. import android.hardware.SensorEventList
ener;
24. import android.hardware.SensorManager;
25. import android.media.MediaPlayer;
26. import android.net.Uri;
27. import android.os.Bundle;
28. import android.os.Environment;
29. import android.os.Handler;
30. import android.os.Message;
31. import android.text.format.Time;
32. import android.util.Log;
33. import android.view.Gravity;
34. import android.view.KeyEvent;
35. import android.view.LayoutInflater;
36. import android.view.View;
37. import android.view.View.OnClickListener;
38. import android.view.ViewGroup;
39. import android.widget.AdapterView;
40. import android.widget.AdapterView;
41. import android.widget.Button;
42. import android.widget.LinearLayout;

```

```

43. import android.widget.LinearLayout.Layo
    utParams;
44. import android.widget.ListView;
45. import android.widget.TabHost;
46. import android.widget.TabHost.TabSpec;
47. import android.widget.TextView;
48. import android.widget.Toast;
49. import org.apache.http.HttpResponse;
50. import org.apache.http.HttpStatus;
51. import org.apache.http.NameValuePair;
52. import org.apache.http.client.ResponseH
    andler;
53. import org.apache.http.client.entity.Ur
    lEncodedFormEntity;
54. import org.apache.http.client.methods.H
    ttpGet;
55. import org.apache.http.client.methods.H
    ttpPost;
56. import org.apache.http.impl.client.Defa
    ultHttpClient;
57. import org.apache.http.message.BasicNam
    eValuePair;
58. import org.apache.http.util.EntityUtil
    s;
59. import java.io.BufferedReader;
60. import java.io.FileReader;
61. import java.io.IOException;
62. import java.io.InputStream;
63. import java.io.OutputStream;
64. import java.io.UnsupportedEncodingException;
65. import java.math.BigDecimal;
66. import java.net.URI;
67. import java.net.URISyntaxException;
68. import java.util.ArrayList;
69. import java.util.Calendar;
70. import java.util.List;
71. import java.util.Set;
72. import java.util.UUID;
73. import java.util.Vector;
74. public class Kanna2Activity extends Act
    ivity implements SensorEventListener {
75.     private float y_sensor;
76.     private float x_sensor;
77.     private float goukei;
78.     private float goukei_x;
79.     private String str_rank;
80.     private float y_SensorValue;
81.     private float x_kosei;
82.     private float x_button;
83.     private float x_kosei2;
84.     private int maxInt = 0;
85.     private int minInt = 0;
86.     private int avarageInt_kanna;
87.     private int divInt_kanna;
88.     private boolean divflag_kanna = false;
89.     private boolean startFlag;
90.     private SharedPreferences modelpref;
91.     private long start;
92.     private long stop;
93.     private long alltime;
94.     private Runnable looper;
95.     private Time time;
96.     private String date;
97.     private String date2;
98.     private int toggle;
99.     private int point;
100.    private int[] currentlyInt_kanna;
101.    private int timesInt;
102.    private int REPEAT_INTERVAL;
103.    private boolean isRepeat;
104.    private Thread thread;
105.    private int wide;
106.    private int hight;
107.    private Bitmap bmp = null;
108.    private Bitmap bmpA = null;
109.    private Bitmap bmp_model = null;
110.    private Bitmap bmp_modelA = null;
111.    private Canvas bmpCanvas;
112.    private Canvas bmpCanvasA;
113.    private TextView textV;
114.    private TextView countTextV;
115.    private Button countBtn;
116.    private LinearLayout countLayout;
117.    private float mYOffset;
118.    private float mYOffsetA;
119.    private MediaPlayer mp = null;
120.    Handler mHandler = new Handler();
121.    Handler dHandler = new Handler();
122.    SensorCalibration SC = new SensorCalibr
        ation();
123.    String modeldata_str;
124.    String strint;
125.    String modelName;
126.    String logsttr;
127.    int jikan;
128.    private Thread threads;
129.    String result;
130.    boolean finishflg = false;
131.    private SensorManager sensorManager;
132.    float gyro_value[];
133.    boolean isKannaMove = false;
134.    boolean graphKannaMove = false;
135.    int modelZureInt;
136.    int zureInt;
137.    float modelBure;
138.    Paint paintSensor = new Paint();
139.    Paint paintLine = new Paint();
140.    Paint paintLine2 = new Paint();
141.    Paint paintBorder = new Paint();
142.    Paint paintText = new Paint();
143.    String Data = null;
144.    final int KANNA = 1;
145.    ArrayList<Integer> docchi = new ArrayLi
        st<Integer>();
146.    LinearLayout btnLayout;
147.    Button okBtn;
148.    Button endBtn;
149.    Button replayBtn;
150.    private float xGraph = 20;
151.    private float pastY = 200;
152.    private float pastYs = 200;
153.    private float pastYA = 200;
154.    float gyroValue;
155.    ArrayList<Float> bureList = new ArrayLi
        st<Float>();
156.    float bureBorder1 = 70;
157.    float bureBorder2 = 90;
158.    float speedBorder1 = 70;
159.    float speedBorder2 = 90;
160.    int intCount = 4;
161.    String timing = "シンプル版のため結果が表
        示されません。フル版では腰の動きも含めてセン
        シング・フィードバックすることができます。
        ";
162.    long kanno_time = 0;
163.    AlertDialog.Builder alertDialogBuilder;
164.    String username;
165.    ArrayList<String> arrayName;
166.    ArrayList<String> arrayGakuban;
167.    ArrayList<String> arrayKeyId;
168.    static ArrayAdapter<String> adapter;
169.    Editor e;
170.    android.app.AlertDialog.Builder dialog
        = null;
171.    AlertDialog dialog2;

```

```

172. ArrayList<Float> speedList = new ArrayL
ist<Float>();
173. ArrayList<Float> blurList = new ArrayLi
st<Float>();
174. ArrayList<String> saveList = new ArrayL
ist<String>();
175. float postSpeed;
176. float postBlur;
177. int speedRank;
178. int blurRank;
179. final int A = 3;
180. final int B = 2;
181. final int C = 1;
182. String speedRankStr;
183. float[] acc = new float[3];
184. float[] gyr = new float[3];
185. long unixTime;
186. long t;
187. float y_hosei;
188. float z_hosei;
189. float Acc;
190. float Acckako;
191. float Spa;
192. float Vel;
193. float kakoVel;
194. float hosei = (float) Math.cos(Math.PI
/ 4f);
195. int kessonchi = 0;
196. Path path = new Path();
197. Path pathA = new Path();
198. float moveLine = 3;
199. int graphDownReset = 5;
200. int graphDownInt = graphDownReset;
201. ArrayList<Float> Ylist = new ArrayList<
Float>();
202. ArrayList<Float> YlistA = new ArrayList
<Float>();
203. ArrayList<Float> YlistA2 = new ArrayLis
t<Float>();
204. ArrayList<Float> Ylist2 = new ArrayList
<Float>();
205. int XYnum = 0;
206. VgraphView VgraphView;
207. AgraphView AgraphView;
208. AlertDialog.Builder replayDlg;
209. ReplayView replayView;
210. int replayHeight = 0;
211. int replayWidth = 0;
212. float replayRight = 0;
213. float replayCenter = 0;
214. float replayLeft = 0;
215. float replayLength = 0;
216. float replayRatio = 0;
217. float replayPosition = 0;
218. float model_replayPosition = 0;
219. Bitmap kannu;
220. Button replayBtn2;
221. Runnable replayer;
222. Thread replay;
223. ArrayList<Float> model_Ylist = new Arra
yList<Float>();
224. View resultView;
225. String ViewStr;
226. int userClassNum = 2;
227. String ID = "";
228. int speedPoint = 0;
229. int burePoint = 0;
230. int koshiPoint = 0;
231. int totalPoint = 0;
232. int userId;
233. float span = 0;
234. ProgressDialog progressDialog;
235. BluetoothAdapter mAdapter;
236. Vector<BluetoothDevice> mDevices;
237. ServerThread serverThread;
238. private boolean divflag_koshi = false;
239. private float div_koshi;
240. long koshi_time = 0;
241. long koshi_time_int = 0;
242. boolean koshiPlot = false;
243. final int KOSHI = 0;
244. int practiceNo = 1;
245. ArrayList<Float> koshiList = new ArrayL
ist<Float>();
246. String saveBure;
247. float docchif = 0;
248. String saveKoshi;
249. String saveTime;
250. String saveText;
251. String saveHosei;
252. Calendar cld;
253. final String NAME = "BLUETOOTH_SAMPLE";
254. UUID MY_UUID = UUID.fromString("1111111
-0000-1000-1111-00AAEECCAFFF");
255. BluetoothSocket[] socket = new Bluetoot
hSocket[1];
256. ConnectedThread[] connection = new Conn
ectedThread[1];
257. String answers = "";
258. String sesskey = "";
259. String attempt = "";
260. String usageid = "";
261. int speedAnswer = 0;
262. int bureAnswer = 0;
263. int koshiAnswer = 0;
264. String moodlehost = "<host>";
265. @Override
266. public void onCreate(Bundle savedInstan
ceState) {
267. super.onCreate(savedInstanceState);
268. Intent intent = getIntent();
269. sesskey = intent.getStringExtra("sesske
y");
270. attempt = intent.getStringExtra("attemp
t");
271. usageid = intent.getStringExtra("usagei
d");
272. sensorManager = (SensorManager) getSyst
emService(Context.SENSOR_SERVICE);
273. gyro_value = new float[4];
274. AlertDialog.Builder a = new AlertDialog.
Builder(this);
275. a.setMessage("モデルが選択されていま
せん");
276. a.setPositiveButton("OK", new DialogInt
erface.OnClickListener() {
277. public void onClick(DialogInterface dia
log, int id) {
278. finish();
279. }
280. });
281. e = getSharedPreferences("2modelpref",
MODE_WORLD_READABLE | MODE_WORLD_WRITEA
BLE).edit();
282. e.putString("username", "");
283. e.putString("userid", "");
284. e.commit();
285. textV = new TextView(this);
286. textV.setText("ここに結果が表示されま
す");
287.
288. VgraphView = new VgraphView(this);
289. AgraphView = new AgraphView(this);
290. btnLayout = new LinearLayout(this);
291. btnLayout.setGravity(Gravity.BOTTOM | G
ravity.CENTER_HORIZONTAL);
292. okBtn = new Button(this);

```

```

293. endBtn = new Button(this);
294. replayBtn = new Button(this);
295. countLayout = new LinearLayout(this);
296. countLayout.setOrientation(LinearLayo
t.VERTICAL);
297. countLayout.setBackgroundResource(<pac
kageName>.R.drawable.start);
298. setContentView(countLayout);
299. countTextV = new TextView(this);
300. countTextV.setText("↓");
301. countTextV.setTextColor(Color.WHITE);
302. countTextV.setTextSize(200);
303. countTextV.setGravity(Gravity.CENTER |
Gravity.CENTER);
304. countBtn = new Button(this);
305. countBtn.setBackgroundResource(<packag
eName>.R.drawable.go_xml);
306. countBtn.setEnabled(false);
307. countLayout.addView(countBtn, new Linea
rLayout.LayoutParams(ViewGroup.LayoutParams
.FILL_PARENT, 250));
308. countLayout.addView(countTextV, new Lin
earLayout.LayoutParams(
ViewGroup.LayoutParams.FILL_PARENT,
ViewGroup.LayoutParams.FILL_PARENT));
311. paintSensor.setColor(Color.CYAN);
312. paintSensor.setStyle(Paint.Style.STROK
E);
313. paintSensor.setStrokeWidth(4);
314. paintSensor.setAntiAlias(true);
315. paintLine.setColor(Color.WHITE);
316. paintLine.setStyle(Paint.Style.FILL);
317. paintLine.setStrokeWidth(3);
318. paintLine2.setColor(Color.WHITE);
319. paintLine2.setStyle(Paint.Style.FILL);
320. paintLine2.setStrokeWidth(1);
321. paintBorder.setColor(Color.WHITE);
322. paintBorder.setStyle(Paint.Style.FIL
L);
323. paintBorder.setStrokeWidth(30);
324. paintText.setColor(Color.CYAN);
325. paintText.setStyle(Paint.Style.FILL);
326. paintText.setStrokeWidth(2);
327. paintText.setTextSize(30);
328. replayView = new ReplayView(this);
329. kanna = BitmapFactory.decodeResource(ge
tResources(), <packageName>.R.drawable.k
anna);
330. LayoutInflater inflater = LayoutInflate
r.from(this);
331. resultView = inflater.inflate(<packageN
ame>.R.layout.result_view, null);
332. TabHost host = (TabHost) resultView.fin
dViewById(<packageName>.R.id.tabhost);
333. host.setup();
334. TabSpec tabComment = host.newTabSpec("C
omment");
335. tabComment.setIndicator("コメント");
336. tabComment.setContent(<packageName>.R.
id.resultComment);
337. host.addTab(tabComment);
338. TabSpec tabGraph = host.newTabSpec("Gra
ph");
339. tabGraph.setIndicator("グラフ");
340. tabGraph.setContent(<packageName>.R.id.
resultGraph);
341. host.addTab(tabGraph);
342. TabSpec tabReplay = host.newTabSpec("Re
play");
343. tabReplay.setIndicator("リプレイ");
344. tabReplay.setContent(<packageName>.R.i
d.resultReplay);
345. host.addTab(tabReplay);

```

```

346. countBtn.setOnClickListener(new View.On
ClickListener() {
347. public void onClick(View v) {
348. graphKannaMove = true;
349. mp = MediaPlayer.create(Kanna2Activity.
this, <packageName>.R.raw.button);
350. try {
351. mp.prepare();
352. } catch (IllegalStateException e) {
353. e.printStackTrace();
354. } catch (IOException e) {
355. e.printStackTrace();
356. }
357. mp.start();
358. mp = MediaPlayer.create(Kanna2Activity.
this, <packageName>.R.raw.count);
359. try {
360. mp.prepare();
361. } catch (IllegalStateException e) {
362. e.printStackTrace();
363. } catch (IOException e) {
364. e.printStackTrace();
365. }
366. mp.start();
367. Countdown();
368. countBtn.setVisibility(View.GONE);
369. countTextV.setTextSize(300);
370. }
371. });
372. modelpref = getSharedPreferences("2mode
lpref", MODE_WORLD_READABLE
| MODE_WORLD_WRITEABLE);
373. modeldata_str = modelpref.getString("2m
odelint", "");
374. strint = modelpref.getString("2modelint
", "");
375. modelName = modelpref.getString("2name",
"");
377. if (modelName != null) {
378. if (!modelName.equals("")) {
379. modelZureInt = Integer.parseInt(modelpr
ef.getString(modelName + "2zure", ""));
380. logsttr = modelpref.getString(modelName
+ "2maxInt", "");
381. modelBure = Marumeru(Float.parseFloat(m
odelpref.getString(modelName + "2maxBur
e", "")), 2);
382. username = modelpref.getString("usernam
e", "");
383. BufferedReader br = null;
384. StringBuilder sb = new StringBuilder();
385. try {
386. try {
387. br = new BufferedReader(new FileReader
(Environment
.getExternalStorageDirectory().getPath
())
+ "/"
+ getPackageName()
+ "/two/"
+ modelName
+ ".txt");
394. String str;
395. while ((str = br.readLine()) != null) {
396. sb.append(str);
397. }
398. } finally {
399. if (br != null) {
400. br.close();
401. }
402. }
403. } catch (IOException e) {
404. }

```

```

405. String allData = sb.toString();
406. String rawData[] = allData.split(",");
407. for (int i = 0; i < rawData.length; i+
+) {
408. try {
409. model_Ylist.add(Float.parseFloat(rawDa
ta[i]));
410. replayLength += Float.parseFloat(rawDat
a[i]);
411. } catch (NumberFormatException e1) {
412. model_Ylist.add(0f);
413. }
414. }
415. replayRatio = (replayHeight - 50) / rep
layLength;
416. } else {
417. a.show();
418. }
419. } else {
420. a.show();
421. }
422. mDevices = new Vector<BluetoothDevice>
();
423. mAdapter = BluetoothAdapter.getDefaultA
dapter();
424. Set<BluetoothDevice> devices = mAdapter.
getBondedDevices();
425. for (BluetoothDevice device : devices)
{
426. mDevices.add(device);
427. serverThread = new ServerThread();
428. serverThread.start();
429. }
430. Runnable countRunnable;
431. public void CountDown() {
432. final int REPEAT_INTERVAL = 1000;
433. final Handler handler = new Handler();
434. countRunnable = new Runnable() {
435. public void run() {
436. intCount--;
437. countTextView.setText(String.valueOf(intC
ount));
438. if (intCount == 0) {
439. handler.removeCallbacks(countRunnabl
e);
440. countTextView.setTextSize(150);
441. countTextView.setText("始¥nめ¥n!");
442. onBtn();
443. } else {
444. handler.postDelayed(this, REPEAT_INTERV
AL);
445. }
446. }
447. };
448. handler.postDelayed(countRunnable, REPE
AT_INTERVAL);
449. }
450. public void onBtn() {
451. currentlyInt_kanna = new int[10];
452. mHandler = new Handler();
453. startFlag = true;
454. isRepeat = false;
455. REPEAT_INTERVAL = 5;
456. time = new Time("Asia/Tokyo");
457. looper = new Runnable() {
458. public void run() {
459. while (isRepeat) {
460. try {
461. Thread.sleep(REPEAT_INTERVAL);
462. } catch (InterruptedException e) {
463. }
464. mHandler.post(new Runnable() {
465. public void run() {
466. doSomething();
467. }
468. });
469. }
470. }
471. };
472. isRepeat = true;
473. goukei = 0;
474. goukei_x = 0;
475. start = 0;
476. stop = 0;
477. maxInt = 0;
478. minInt = 0;
479. timesInt = 0;
480. start = System.currentTimeMillis();
481. time.setToNow();
482. thread = new Thread(looper);
483. thread.start();
484. textView.setTextSize(75);
485. textView.setTextColor(Color.WHITE);
486. textView.setText("計測中...");
487. textView.setPadding(0, 100, 0, 0);
488. textView.setGravity(Gravity.CENTER);
489. float y_btn = y_sensor * 10;
490. y_btn = Math.round(y_btn);
491. y_btn = y_btn / 10;
492. SC.setBtnSensorValue(y_btn);
493. x_kosei = x_sensor * 10;
494. x_kosei = Math.round(x_kosei);
495. x_kosei = x_kosei / 10;
496. x_button = x_kosei;
497. }
498. private void doSomething() {
499. jikan++;
500. gyro_value[3] = gyro_value[3] + Math.ab
s(gyro_value[0]) + Math.abs(gyro_value
[1]) + Math.abs(gyro_value[2]);
501. int finishjikan = jikan / 10;
502. y_SensorValue = SC.setNowSensorValue(y_
sensor);
503. x_kosei = x_sensor * 10;
504. x_kosei = Math.round(x_kosei);
505. x_kosei = x_kosei / 10;
506. if (startFlag == true) {
507. timesInt++;
508. for (int i = 9; i >= 1; i--) {
509. currentlyInt_kanna[i] = currentlyInt_ka
nna[i - 1];
510. }
511. currentlyInt_kanna[0] = (int) y_SensorV
alue;
512. avarageInt_kanna = (currentlyInt_kanna
[0] + currentlyInt_kanna[1]
+ currentlyInt_kanna[2] + currentlyInt_
kanna[3]
+ currentlyInt_kanna[4] + currentlyInt_
kanna[5]
+ currentlyInt_kanna[6] + currentlyInt_
kanna[7]
+ currentlyInt_kanna[8] + currentlyInt_
kanna[9]) / 10;
517. int[] deviationInt_kanna = new int[10];
518. float[] deviation_koshi = new float[1
0];
519. for (int i = 0; i <= 9; i++) {
520. deviationInt_kanna[i] = Math.abs(curren
tlyInt_kanna[i]
- avarageInt_kanna);
521. }
522. }
523. divInt_kanna = (deviationInt_kanna[0] +
deviationInt_kanna[1]
+ deviationInt_kanna[2] + deviationInt_
kanna[3]

```

```

525. + deviationInt_kanna[4] + deviationInt_
kanna[5]
526. + deviationInt_kanna[6] + deviationInt_
kanna[7]
527. + deviationInt_kanna[8] + deviationInt_
kanna[9]) / 10;
528. if (divflag_koshi == false && div_koshi
> 0.7f) {
529. koshi_time = System.currentTimeMillis()
-75;
530. divflag_koshi = true;
531. koshiPlot = true;
532. docchi.add(KOSHI);
533. koshi_time_int = jikan;
534. }
535. if (divflag_kanna == false && divInt_ka
nna > 0) {
536. divflag_kanna = true;
537. isKannaMove = true;
538. kanna_time = System.currentTimeMillis
();
539. docchi.add(KANNA);
540. }
541. if (finishjikan == 36) {
542. divflag_kanna = false;
543. finishflg = true;
544. countTextV.setText("分¥n析¥n中");
545. dostop();
546. }
547. bureList.add(gyroValue);
548. if (maxInt < -y_SensorValue) {
549. maxInt = (int) -y_SensorValue;
550. }
551. if (minInt > -y_SensorValue) {
552. minInt = (int) -y_SensorValue;
553. }
554. goukei = goukei + Math.abs(y_SensorValu
e);
555. x_kosei2 = ((x_kosei) * 10 - (x_button)
* 10) / 10;
556. goukei_x = goukei_x + Math.abs(x_kosei
2);
557. if (jikan == 1) {
558. unixTime = System.currentTimeMillis();
559. AccKako = Acc;
560. } else {
561. t = System.currentTimeMillis() - unixTi
me;
562. unixTime = System.currentTimeMillis();
563. Acc = -acc[1];
564. if (t > 10) {
565. kessonchi = 5;
566. } else {
567. if (kessonchi > 0) {
568. kessonchi--;
569. if (Acc <= 0.5f && Acc >= -0.5f) {
570. Spa = AccKako * t;
571. YlistA.add(Marumeru(AccKako,1));
572. } else {
573. Spa = Acc * t;
574. AccKako = Acc;
575. YlistA.add(Marumeru(Acc,1));
576. }
577. } else {
578. Spa = Acc * t;
579. YlistA.add(Marumeru(Acc,1));
580. }
581. kakoVel = Vel;
582. Vel = Vel + Spa;
583. Ylist.add(Vel);
584. }
585. }
586. saveList.add(username + "," + practiceN
o + ","
587. + System.currentTimeMillis() + "," + ac
c[0] + "," + acc[1]
588. + "," + acc[2] + "," + gyr[0] + "," + g
yr[1] + "," + gyr[2]
589. + "," + Data + ","");
590. blurList.add(gyroValue);
591. speedList.add(postSpeed);
592. koshiList.add(Float.valueOf(Data));
593. VgraphView.invalidate();
594. AgraphView.invalidate();
595. }
596. }
597. private void dostop() {
598. isRepeat = false;
599. pastY = pastYs;
600. xGraph = 20;
601. boolean isGraphStart = false;
602. boolean isGraphZero = false;
603. float max = 0;
604. int maxLim = 3;
605. boolean isMax = false;
606. float y2 = 0;
607. float yA2 = 0;
608. float henkyoku = 0;
609. int henkyokuten = 0;
610. float kyokusho = 1000000;
611. float kyokushoA = 0;
612. int kyokushoten = 0;
613. float down = 0;
614. int zeroten = 0;
615. for (int i = 0; i < Ylist.size(); i++)
{
616. y2 = Ylist.get(i);
617. if (max <= y2) {
618. max = y2;
619. maxLim = 3;
620. }
621. if (!isGraphStart) {
622. zureInt++;
623. }
624. if (!isGraphStart && y2 >= 250) {
625. isGraphStart = true;
626. }
627. if (isGraphStart && max > y2) {
628. maxLim--;
629. }
630. if (maxLim == 0) {
631. isMax = true;
632. }
633. if (isMax && i > 0) {
634. if (henkyoku <= y2 - Ylist.get(i - 1))
{
635. henkyoku = y2 - Ylist.get(i - 1);
636. henkyokuten = i - 7;
637. }
638. }
639. if (isMax && kyokusho >= y2) {
640. kyokusho = y2;
641. kyokushoten = i;
642. kyokushoA = yA2;
643. }
644. if (!isGraphZero && isMax && y2 <= 0) {
645. isGraphZero = true;
646. zeroten = i;
647. }
648. }
649. down = kyokushoten - henkyokuten;
650. if (down < 0) {
651. down = 7;
652. }
653. for (int i = 0; i < Ylist.size(); i++)
{

```

```

654. y2 = Ylist.get(i);
655. yA2 = YlistA.get(i);
656. if (isGraphZero) {
657.   if (i >= zeroten) {
658.     y2 = 0;
659.     yA2 = 0;
660.   }
661. } else {
662.   if (i >= kyokushoten) {
663.     y2 = 0;
664.     yA2 = 0;
665.   } else if (i >= henkyokuten) {
666.     y2 = y2 - (((i - henkyokuten) / down) *
        kyokusho);
667.     yA2 = yA2 + (((i - henkyokuten) / down)
        * kyokushoA);
668.   }
669. }
670. Ylist2.add(y2);
671. YlistA2.add(yA2);
672. replayPosition = 0;
673. model_replayPosition = 0;
674. }
675. threads = null;
676. stop = System.currentTimeMillis();
677. alltime = (stop - start) / 1000;
678. if (alltime == 0) {
679.   alltime = 1;
680. }
681. float speedMax = Float.parseFloat(Maxim
um(Ylist2)[0]);
682. int modelSpeed = (int) Float.parseFloat
(logsttr);
683. speedPoint = (int) ((100 * speedMax) /
modelSpeed);
684. if (speedPoint > 100) {
685.   speedPoint = 100;
686. }
687. if (speedPoint < speedBorder1) {
688.   speedRank = C;
689.   speedRankStr = "★☆☆";
690. } else if (speedPoint >= speedBorder2)
{
691.   speedRank = A;
692.   speedRankStr = "★★★";
693. } else {
694.   speedRank = B;
695.   speedRankStr = "★★☆";
696. }
697. speedAnswer = speedPoint;
698. String[] bureResult = Maximum(bureLis
t);
699. float bure = Float.parseFloat(bureResul
t[0]);
700. burePoint = 0;
701. if (bure == 0f) {
702.   burePoint = 100;
703. } else {
704.   burePoint = (int) (Marumeru(((100 * mod
elBure) / bure), 0));
705. }
706. if (burePoint > 100) {
707.   burePoint = 100;
708. }
709. if (burePoint < bureBorder1) {
710.   blurRank = C;
711.   str_rank = "★☆☆";
712. } else if (burePoint >= bureBorder2) {
713.   blurRank = A;
714.   str_rank = "★★★";
715. } else {
716.   blurRank = B;
717.   str_rank = "★★☆";
718. }
719. saveBure = String.valueOf(bure);
720. bureAnswer = burePoint;
721. if (speedRank == A) {
722.   if (blurRank == A) {
723.     result = "ブレもなく、良いスピードで引くこ
とができています。その調子!";
724.   } else if (blurRank == B) {
725.     result = "良いスピードで引けていますが、少
しブレがあります。もう少し丁寧に引いてみま
しょう。";
726.   } else if (blurRank == C) {
727.     result = "良いスピードで引けていますが、ブ
レが大きいです。力任せに引くのではなく、丁
寧さも心掛けましょう。";
728.   }
729. } else if (speedRank == B) {
730.   if (blurRank == A) {
731.     result = "ブレることなく丁寧に引けています。
次はそのままもう少し速く引いてみましょう。
";
732.   } else if (blurRank == B) {
733.     result = "少しかんながブレているようです。
丁寧に、かつもう少し速く引いてみましょう。";
734.   } else if (blurRank == C) {
735.     result = "かんなのブレが大きいです。もっと
丁寧に、かつもう少し速く引いてみましょう。";
736.   }
737. } else if (speedRank == C) {
738.   if (blurRank == A) {
739.     result = "丁寧に引くことができていますが、
少し遅いです。もっと速く引いてみましょう。";
740.   } else if (blurRank == B) {
741.     result = "引く速度が遅いです。少しブレがあ
るので、丁寧に忘れないようにしつつ、もっ
と速く引いてみましょう。";
742.   } else if (blurRank == C) {
743.     result = "ブレが大きく、引く速度も遅いです。
丁寧に忘れないようにしつつ、もっと速く引
いてみましょう。";
744.   }
745. }
746. if (docchi.size() == 2) {
747.   docchif = (kanna_time - koshi_time) / 1
000f;
748.   if (docchif > 0) {
749.     docchif = (kanna_time - koshi_time) / 1
000f;
750.     timing = "腰から全身でしっかり引くことがで
きています!¥n ランク:★★★";
751.     saveKoshi = "KOSHI->TE";
752.     koshiPoint = 100;
753.     koshiAnswer = 3;
754.   } else {
755.     timing = "手元が先に動いています。腰から全
身を使って引きましょう!¥n ランク:★★☆
";
756.     saveKoshi = "TE->KOSHI";
757.     koshiPoint = 70;
758.     koshiAnswer = 2;
759.   }
760.   saveTime = String.valueOf(docchif);
761. } else {
762.   timing = "上半身だけで引いています。腰から
全身で引きましょう!¥n ランク:★☆☆";
763.   saveKoshi = "TE only";
764.   saveTime = "-";
765.   koshiPoint = 30;
766.   koshiAnswer = 1;
767. }
768. ViewStr = "¥n【引きの速さとブレ】" + "¥n
師匠を100点とすると…" + "¥n あなたの引き
の速さは" + speedPoint

```

```

769. + "点です。" + "%n プレの少なさは" + bure
Point + "点です。" + "%n 速さランク："
770. + speedRankStr + "%n プレランク：" + str
_rank + "%n" + result
771. + "%n%n 【全身の使い方】" + "%n" + timing;
772. textV.setText(Color.WHITE);
773. textV.setText("計測完了");
774. textV.setText(ViewStr);
775. textV.setTextSize(17);
776. textV.setPadding(0, 10, 0, 0);
777. textV.setGravity(Gravity.LEFT);
778. startFlag = false;
779. answers = speedAnswer + "," + bureAnsw
er + "," + koshiAnswer;
780. mp = MediaPlayer.create(this, <packageN
ame>.R.raw.result);
781. try {
782. mp.prepare();
783. } catch (IllegalStateException e) {
784. e.printStackTrace();
785. } catch (IOException e) {
786. e.printStackTrace();
787. }
788. mp.start();
789. saveText = "";
790. for (int i = 0; i < saveList.size(); i+
+) {
791. saveText += "%n" + saveList.get(i);
792. if (i == saveList.size() - 1) {
793. saveText += saveKoshi + "," + saveTime;
794. }
795. }
796. saveHosei = "";
797. for (int i = 0; i < Ylist2.size(); i++)
{
798. saveHosei += "%n" + username + "," + pr
acticeNo + "," + Ylist2.get(i) + "," +
YlistA2.get(i) + "," + burelist.get(i);
799. }
800. saveHosei += "," + speedPoint + "," + s
peedRank + "," + burePoint + "," + blur
Rank + "," + saveKoshi;
801. LayoutInflater inflater = LayoutInflate
r
802. .from(Kanna2Activity.this);
803. resultView = inflater.inflate(<packageN
ame>.R.layout.result_view, null);
804.
805. TabHost host = (TabHost) resultView.fin
dViewById(<packageName>.R.id.tabhost);
806. host.setup();
807. TabSpec tabComment = host.newTabSpec("C
omment");
808. tabComment.setIndicator("コメント");
809. tabComment.setContent(<packageName>.R.
id.resultComment);
810. host.addTab(tabComment);
811. TabSpec tabGraph = host.newTabSpec("Gra
ph");
812. tabGraph.setIndicator("グラフ");
813. tabGraph.setContent(<packageName>.R.id.
resultGraph);
814. host.addTab(tabGraph);
815. TabSpec tabReplay = host.newTabSpec("Re
play");
816. tabReplay.setIndicator("リプレイ");
817. tabReplay.setContent(<packageName>.R.i
d.resultReplay);
818. host.addTab(tabReplay);
819. replayView = new ReplayView(Kanna2Activ
ity.this);
820. replayBtn2 = new Button(Kanna2Activity.
this);
821. replayBtn2.setText("リプレイ");
822. replayBtn2.setLayoutParams(new LayoutPa
rams(
823. LayoutParams.WRAP_CONTENT, LayoutParams.
WRAP_CONTENT));
824. replayBtn2.setOnClickListener(new OnCli
ckListener() {
825. public void onClick(View v) {
826. replayPosition = 0;
827. model_replayPosition = 0;
828. replayer = new Runnable() {
829. public void run() {
830. mHandler.post(new Runnable() {
831. public void run() {
832. replayView.invalidate();
833. }
834. });
835. try {
836. Thread.sleep(500);
837. } catch (InterruptedException e) {
838. }
839. for (int i = zureInt; i < Ylist2.size()
|| modelZureInt + i - zureInt < model_Y
list
840. .size(); i++) {
841. if (i < Ylist2.size()) {
842. replayPosition += Ylist2.get(i);
843. } else {
844. replayPosition += 0;
845. }
846. }
847. if (modelZureInt + i - zureInt < model_
Ylist
848. .size()) {
849. model_replayPosition += model_Ylist
.get(modelZureInt + i - zureInt);
850. } else {
851. model_replayPosition += 0;
852. }
853. }
854. try {
855. Thread.sleep(REPEAT_INTERVAL * 5);
856. } catch (InterruptedException e) {
857. }
858. mHandler.post(new Runnable() {
859. public void run() {
860. replayView.invalidate();
861. }
862. });
863. }
864. }
865. };
866. replay = new Thread(replayer);
867. replay.start();
868. }
869. });
870. replayDlg = new AlertDialog.Builder(Kan
na2Activity.this);
871. replayDlg.setView(resultView);
872. VgraphView = new VgraphView(Kanna2Activ
ity.this);
873. AgraphView = new AgraphView(Kanna2Activ
ity.this);
874. TextView rt = (TextView) resultView
.findViewById(<packageName>.R.id.resul
tText);
875. LinearLayout g = (LinearLayout) resultV
iew
876. .findViewById(<packageName>.R.id.resul
tGraph);
877. LinearLayout r = (LinearLayout) resultV
iew
878. .findViewById(<packageName>.R.id.resul
tReplay);
879. rt.setText(ViewStr);
880.

```

```

881. rt.setTextSize(17);
882. rt.setTextColor(Color.WHITE);
883. LinearLayout.LayoutParams lp = new Line
arLayoutParams(
884.   LinearLayout.LayoutParams.FILL_PARENT, LinearLayout.
WRAP_CONTENT);
885. lp.weight = 1;
886. g.addView(VgraphView, lp);
887. g.addView(AgraphView, lp);
888. r.setGravity(Gravity.CENTER_HORIZONTA
L);
889. r.addView(replayBtn2);
890. r.addView(replayView);
891. replayDlg.setCancelable(false);
892. replayDlg.setPositiveButton(" もう一度
",
893.   new DialogInterface.OnClickListener() {
894.     public void onClick(DialogInterface dia
log, int id) {
895.       intCount = 4;
896.       jikan = 0;
897.       zureInt = 1;
898.       pastY = pastYs;
899.       pastYA = pastYs;
900.       xGraph = 20;
901.       countBtn.setVisibility(View.VISIBLE);
902.       Paint p = new Paint();
903.       p.setColor(Color.WHITE);
904.       if (bmpCanvas != null && bmpCanvasA !=
null) {
905.         bmpCanvas.drawColor(Color.TRANSPARENT,
PorterDuff.Mode.CLEAR);
906.         bmpCanvasA.drawColor(Color.TRANSPARENT,
PorterDuff.Mode.CLEAR);
907.       }
908.       countTextV.setText(" ↓");
909.       countTextV.setTextSize(200);
910.       divflag_kanna = false;
911.       currentlyInt_kanna = new int[10];
912.       docchi.clear();
913.       bureList.clear();
914.       speedList.clear();
915.       blurList.clear();
916.       saveList.clear();
917.       unixTime = 0;
918.       t = 0;
919.       y_hosei = 0;
920.       z_hosei = 0;
921.       Acc = 0;
922.       AccKako = 0;
923.       Spa = 0;
924.       Vel = 0;
925.       kakoVel = 0;
926.       kessonchi = 0;
927.       graphDownInt = graphDownReset;
928.       path = new Path();
929.       pathA = new Path();
930.       XYnum = 0;
931.       Ylist.clear();
932.       Ylist2.clear();
933.       YlistA.clear();
934.       YlistA2.clear();
935.     }
936.   });
937. replayDlg.setNegativeButton(" 練習を終わ
る",
938.   new DialogInterface.OnClickListener() {
939.     public void onClick(DialogInterface dia
log, int id) {
940.       Uri uri =Uri.parse( moodlehost + "quest
ion/type/motorskill/posttomoodle.php?a
nswers="+answers+"&sesskey="+sesskey+"
&attempt="+attempt+"&usageid="+usageid
);
941.       Intent intent = new Intent(Intent.ACTIO
N_VIEW, uri);
942.       byte[] postdata = ("answers="+answers+"
&sesskey="+sesskey+"&attempt="+attemp
t).getBytes();
943.       Intent intent2 = intent.putExtra("com.a
ndroid.browser.post_data", postdata);
944.       startActivity(intent2);
945.       finish();
946.     }
947.   });
948. replayDlg.show();
949. }
950. private class VgraphView extends View {
951.   public VgraphView(Context context) {
952.     super(context);
953.     setFocusable(true);
954.   }
955.   protected void onSizeChanged(int w, int
h, int oldw, int oldh) {
956.     super.onSizeChanged(w, h, oldw, oldh);
957.     wide = w;
958.     hight = h;
959.     pastYs = h / 2;
960.     pastY = (hight / 4) * 3;
961.     pastYA = (hight / 4) * 2;
962.     if (bmp == null) {
963.       bmp = Bitmap.createBitmap(w, h, Bitmap.
Config.ARGB_4444);
964.       bmp_model = Bitmap.createBitmap(w, h, B
itmap.Config.ARGB_4444);
965.       bmp_model = BitmapFactory.decodeFile(En
vironment
966.         .getExternalStorageDirectory().getPath
()
967.         + "/"
968.         + getPackageName() + "/two/" + modelNam
e + ".png");
969.     }
970.     bmpCanvas = new Canvas(bmp);
971.     bmpCanvas.drawColor(Color.argb(0, 200,
255, 255));
972.     path.moveTo(20, (hight / 4) * 3);
973.     pathA.moveTo(20, (hight / 4) * 2);
974.     span = (wide-40)/360f;
975.     for (int i = 0; i < Ylist2.size(); i++)
{
976.       mYOffset = MoveAverage(i)[0];
977.       mYOffsetA = MoveAverage(i)[1];
978.       int y = (int) (mYOffset);
979.       int yA = (int) (mYOffsetA);
980.       path.quadTo(xGraph, pastY, (xGraph + xG
raph + span) / 2,
(pastY + y) / 2);
981.       pathA.quadTo(xGraph, pastYA, (xGraph +
xGraph + span) / 2,
(pastYA + yA) / 2);
982.       xGraph = xGraph + span;
983.       pastY = y;
984.       pastYA = yA;
985.     }
986.     bmpCanvas.drawLine(20, hight / 2, wide
- 20, hight / 2, paintLine2);
987.     bmpCanvas.drawLine(20, (hight / 4) * 1,
wide - 20, (hight / 4) * 1,
988.       paintLine2);
989.     bmpCanvas.drawLine(20, (hight / 4) * 3,
wide - 20, (hight / 4) * 3,
990.       paintLine2);
991.     bmpCanvas.drawLine(20, (hight / 4) * 3,
wide - 20, (hight / 4) * 3,
992.       paintLine);
993.     bmpCanvas.drawPath(path, paintSensor);
994.     bmpCanvas.drawCircle(20 + (koshi_time_i
nt - 30) * span, (hight / 4) * 3, 10, p

```

```

aintSensor);
995. bmpCanvas.drawLine(5, hight, wide - 5,
hight, paintBorder);
996. VgraphView.invalidate();
997. }
998. @Override
999. protected void onDraw(Canvas canvas) {
1000. canvas.drawBitmap(bmp, 0, 0, null);
1001. if (isRepeat) {
1002. bmpCanvas.drawLine(20, hight / 2, wide
- 20, hight / 2,
1003. paintLine2);
1004. bmpCanvas.drawLine(20, (hight / 4) * 1,
wide - 20,
1005. (hight / 4) * 1, paintLine2);
1006. bmpCanvas.drawLine(20, (hight / 4) * 3,
wide - 20,
1007. (hight / 4) * 3, paintLine);
1008. if (startFlag == true) {
1009. if (koshiPlot) {
1010. koshiPlot = false;
1011. }
1012. }
1013. bmpCanvas.drawLine(5, hight, wide - 5,
hight, paintBorder);
1014. if (isKannaMove) {
1015. isKannaMove = false;
1016. }
1017. } else {
1018. canvas.drawBitmap(bmp_model, (zureInt-m
odelZureInt)*span, 0, null);
1019. paintText.setColor(Color.WHITE);
1020. paintText.setTextSize(35);
1021. canvas.drawText("速度", 10, 55, paintTe
xt);
1022. paintText.setColor(Color.CYAN);
1023. paintText.setTextSize(25);
1024. canvas.drawText("あなた", 150, 55, pain
tText);
1025. paintText.setColor(Color.YELLOW);
1026. paintText.setTextSize(25);
1027. canvas.drawText("師匠", 250, 55, paintT
ext);
1028. }
1029. }
1030. }
1031. private class AgrahView extends View {
1032. public AgrahView(Context context) {
1033. super(context);
1034. setFocusable(true);
1035. }
1036. protected void onSizeChanged(int w, int
h, int oldw, int oldh) {
1037. super.onSizeChanged(w, h, oldw, oldh);
1038. wide = w;
1039. hight = h;
1040. pastYs = h / 2;
1041. pastY = h / 2;
1042. if (bmpA == null) {
1043. bmpA = Bitmap.createBitmap(w, h, Bitmap.
Config.ARGB_4444);
1044. bmp_modelA = Bitmap.createBitmap(w, h,
Bitmap.Config.ARGB_4444);
1045. bmp_modelA =
1046. BitmapFactory.decodeFile(Environment.g
etExternalStorageDirectory().getPath()
+ "/" + getPackageName() + "/two/" + mo
delName + "A.png");
1047. }
1048. }
1049. bmpCanvasA = new Canvas(bmpA);
1050. bmpCanvasA.drawColor(Color.argb(0, 200,
255, 255));
1051. bmpCanvasA.drawLine(20, hight / 2, wide
- 20, hight / 2, paintLine);
1052. bmpCanvasA.drawLine(20, (hight / 4) * 1,
wide - 20,
1053. (hight / 4) * 1, paintLine2);
1054. bmpCanvasA.drawLine(20, (hight / 4) * 3,
wide - 20,
1055. (hight / 4) * 3, paintLine2);
1056. bmpCanvasA.drawPath(pathA, paintSens
or);
1057. bmpCanvasA.drawCircle(20 + (koshi_time_
int-30)*span, hight / 2, 10, paintSens
or);
1058. bmpCanvasA.drawLine(5, 0, wide - 5, 0,
paintBorder);
1059. AgrahView.invalidate();
1060. }
1061. @Override
1062. protected void onDraw(Canvas canvas) {
1063. canvas.drawBitmap(bmpA, 0, 0, null);
1064. if (isRepeat) {
1065. bmpCanvasA.drawLine(5, 0, wide - 5, 0,
paintBorder);
1066. bmpCanvasA.drawLine(20, hight / 2, wide
- 20, hight / 2,
1067. paintLine);
1068. bmpCanvasA.drawLine(20, (hight / 4) * 1,
wide - 20,
1069. (hight / 4) * 1, paintLine2);
1070. bmpCanvasA.drawLine(20, (hight / 4) * 3,
wide - 20,
1071. (hight / 4) * 3, paintLine2);
1072. } else {
1073. canvas.drawBitmap(bmp_modelA, (zureInt-
modelZureInt)*span, 0, null);
1074. paintText.setTextSize(35);
1075. paintText.setColor(Color.WHITE);
1076. canvas.drawText("加速度", 10, 55, paint
Text);
1077. }
1078. }
1079. }
1080. @Override
1081. public void onResume() {
1082. super.onResume();
1083. Sensor sensor = sensorManager
1084. .getDefaultSensor(Sensor.TYPE_LINEAR_A
CCELERATION);
1085. Sensor sensor2 = sensorManager.getDefau
ltSensor(Sensor.TYPE_GYROSCOPE);
1086. sensorManager.registerListener(this, se
nsor,
1087. SensorManager.SENSOR_DELAY_FASTEST);
1088. sensorManager.registerListener(this, se
nsor2,
1089. SensorManager.SENSOR_DELAY_FASTEST);
1090. }
1091. @Override
1092. public void onPause() {
1093. super.onPause();
1094. sensorManager.unregisterListener(thi
s);
1095. }
1096. public void onSensorChanged(SensorEvent
event) {
1097. switch (event.sensor.getType()) {
1098. case Sensor.TYPE_LINEAR_ACCELERATION:
1099. acc = event.values;
1100. y_sensor = event.values[1];
1101. x_sensor = event.values[0];
1102. postSpeed = SC.setNowSensorValue(-y_sen
sor);
1103. break;
1104. case Sensor.TYPE_GYROSCOPE:

```

```

1105. gyr = event.values;
1106. gyro_value[0] = Math.abs(Marumeru(event.
values[0], 2));
1107. gyro_value[1] = Math.abs(Marumeru(event.
values[1], 2));
1108. gyro_value[2] = Math.abs(Marumeru(event.
values[2], 2));
1109. gyroValue = Marumeru(event.values[1],
2)
1110. + Marumeru(event.values[2], 2);
1111. break;
1112. }
1113. }
1114. public void onAccuracyChanged(Sensor se
nsor, int accuracy) {
1115. }
1116. private class ServerThread extends Thre
ad {
1117. private final BluetoothServerSocket mmS
erverSocket;
1118. public ServerThread() {
1119. BluetoothServerSocket tmp = null;
1120. try {
1121. tmp = mAdapter
1122. .listenUsingRfcommWithServiceRecord(NA
ME, MY_UUID);
1123. } catch (IOException e) {
1124. }
1125. mmServerSocket = tmp;
1126. }
1127. @Override
1128. public void run() {
1129. while (true) {
1130. try {
1131. socket[0] = mmServerSocket.accept();
1132. } catch (IOException e) {
1133. }
1134. if (socket[0] != null) {
1135. manageConnectedSocket(socket[0]);
1136. Message msg = new Message();
1137. msg.obj = "";
1138. toastHandler.sendMessage(msg);
1139. connection[0].start();
1140. }
1141. break;
1142. }
1143. }
1144. @SuppressWarnings("unused")
1145. public void cancel() {
1146. try {
1147. mmServerSocket.close();
1148. } catch (IOException e) {
1149. }
1150. }
1151. }
1152. private class ConnectedThread extends T
hread {
1153. private final BluetoothSocket mmSocket;
1154. private final InputStream mmInStream;
1155. private final OutputStream mmOutStream;
1156. String[] strAry = new String[5];
1157. public ConnectedThread(BluetoothSocket
socket) {
1158. mmSocket = socket;
1159. InputStream tmpIn = null;
1160. OutputStream tmpOut = null;
1161. try {
1162. tmpIn = socket.getInputStream();
1163. tmpOut = socket.getOutputStream();
1164. } catch (IOException e) {
1165. }
1166. mmInStream = tmpIn;
1167. mmOutStream = tmpOut;
1168. }
1169. @Override
1170. public void run() {
1171. byte[] buffer = new byte[1024];
1172. @SuppressWarnings("unused")
1173. int bytes;
1174. while (true) {
1175. try {
1176. bytes = mmInStream.read(buffer);
1177. String readMsg = new String(buffer, "UT
F-8");
1178. Message msg = new Message();
1179. msg.obj = readMsg;
1180. strAry = readMsg.split("ita00");
1181. Data = strAry[0];
1182. } catch (IOException e) {
1183. break;
1184. }
1185. }
1186. }
1187. @SuppressWarnings("unused")
1188. public void write(byte[] bytes) {
1189. try {
1190. mmOutputStream.write(bytes);
1191. } catch (IOException e) {
1192. }
1193. }
1194. @SuppressWarnings("unused")
1195. public void cancel() {
1196. try {
1197. mmSocket.close();
1198. } catch (IOException e) {
1199. }
1200. }
1201. }
1202. public void manageConnectedSocket(Bluet
oothSocket socket) {
1203. connection[0] = new ConnectedThread(soc
ket);
1204. }
1205. final Handler toastHandler = new Handle
r() {
1206. public void handleMessage(Message msg)
{
1207. Toast.makeText(Kanna2Activity.this, msg.
obj + "接続完了",
1208. Toast.LENGTH_SHORT).show();
1209. countBtn.setEnabled(true);
1210. }
1211. };
1212. public boolean onKeyDown(int keyCode, K
eyEvent event) {
1213. if (keyCode == KeyEvent.KEYCODE_BACK) {
1214. System.exit(0);
1215. } else if (keyCode == KeyEvent.KEYCODE_
MENU) {
1216. openOptionsMenu();
1217. }
1218. return true;
1219. }
1220. public float Marumeru(float f, int i) {
1221. BigDecimal xbi = new BigDecimal(String.
valueOf(f));
1222. float result = xbi.setScale(i, BigDecim
al.ROUND_HALF_UP).floatValue();
1223. return result;
1224. }
1225. public String[] Maximum(ArrayList<Float
> list) {
1226. String[] result = new String[3];
1227. float f = 0f;
1228. int index = 0;
1229. String when = "";

```

```

1230. for (int i = 0; i < list.size(); i++) {
1231.     if (f < list.get(i)) {
1232.         f = list.get(i);
1233.         index = i;
1234.     }
1235. }
1236. if (index < list.size() / 3f) {
1237.     when = "引き始め";
1238. } else if (index >= list.size() / 1.5f)
1239.     {
1240.     when = "引き終わり";
1241.     } else {
1242.     when = "引いている途中";
1243.     }
1244. result[0] = String.valueOf(f);
1245. result[1] = String.valueOf(index);
1246. result[2] = when;
1247. return result;
1248. }
1249. public float[] MoveAverage(int i) {
1250.     float[] result = new float[2];
1251.     float Vsum = 0;
1252.     float Asum = 0;
1253.     if (i > 10 && i < Ylist2.size() - 10) {
1254.         for (int j = -5; j < 5; j++) {
1255.             Vsum += Ylist2.get(i + j);
1256.         }
1257.         for (int j = -10; j < 10; j++) {
1258.             Asum += YlistA2.get(i + j);
1259.         }
1260.         result[0] = (hight / 4) * 3 + (-Vsum / 10) / 10;
1261.         result[1] = hight / 2 + (-Asum / 20) * 7;
1262.     } else {
1263.         result[0] = (hight / 4) * 3 + (-Ylist2.get(i)) / 10;
1264.         result[1] = hight / 2 + (-YlistA2.get(i)) * 7;
1265.     }
1266.     return result;
1267. }
1268. public class ReplayView extends View {
1269.     public ReplayView(Context context) {
1270.         super(context);
1271.     }
1272.     protected void onSizeChanged(int w, int h, int oldw, int oldh) {
1273.         super.onSizeChanged(w, h, oldw, oldh);
1274.         replayHeight = h;
1275.         replayWidth = w;
1276.         replayRight = 1 * (replayWidth / 4);
1277.         replayCenter = 2 * (replayWidth / 4);
1278.         replayLeft = 3 * (replayWidth / 4);
1279.         replayRatio = (replayHeight - 50) / replayLength;
1280.     }
1281.     @Override
1282.     protected void onDraw(Canvas canvas) {
1283.         canvas.drawColor(Color.WHITE);
1284.         canvas.drawBitmap(kanna, replayRight, replayPosition * replayRatio, null);
1285.         canvas.drawBitmap(kanna, replayLeft, mo
del_replayPosition
1286.         * replayRatio, null);
1287.         paintText.setTextSize(35);
1288.         paintText.setColor(Color.BLACK);
1289.         canvas.drawText("あなた", replayRight, 100, paintText);
1290.         canvas.drawText("師匠", replayLeft, 100, paintText);
1291.     }
1292. }
1293. public void exec_post(String values) {
1294.     String ret = "";
1295.     URI url = null;
1296.     try {
1297.         url = new URI( moodlehost + "mod/quiz/posttomoodle.php" );
1298.     } catch (URISyntaxException e) {
1299.         e.printStackTrace();
1300.         ret = e.toString();
1301.     }
1302.     HttpPost request = new HttpPost( url );
1303.     List<NameValuePair> post_params = new ArrayList<NameValuePair>();
1304.     post_params.add(new BasicNameValuePair("answers", values));
1305.     post_params.add(new BasicNameValuePair("sesskey", sesskey));
1306.     post_params.add(new BasicNameValuePair("attempt", attempt));
1307.     try {
1308.         request.setEntity(new UrlEncodedFormEntity(post_params, "UTF-8"));
1309.     } catch (UnsupportedEncodingException e1) {
1310.         e1.printStackTrace();
1311.     }
1312.     DefaultHttpClient httpClient = new DefaultHttpClient();
1313.     try {
1314.         ret = httpClient.execute(request, new ResponseHandler<String>() {
1315.             @Override
1316.             public String handleResponse(HttpResponse response) throws IOException
1317.             {
1318.                 switch (response.getStatusLine().getStatusCode()) {
1319.                     case HttpStatus.SC_OK:
1320.                         return EntityUtils.toString(response.getEntity(), "UTF-8");
1321.                     case HttpStatus.SC_NOT_FOUND:
1322.                         return null;
1323.                     default:
1324.                         return null;
1325.                 }
1326.             }
1327.         });
1328.     } catch (IOException e) {
1329.     } finally {
1330.         httpClient.getConnectionManager().shutdown();
1331.     }
1332. }
1333. }

```

2-3 学習者用アプリケーション（腰端末のメインのプログラム）

```

1. package <packageName>;
2. import java.io.IOException;
3. import java.io.InputStream;
4. import java.io.OutputStream;
5. import java.math.BigDecimal;
6. import java.util.List;
7. import java.util.Set;
8. import java.util.UUID;
9. import java.util.Vector;
10. import android.app.Activity;
11. import android.bluetooth.BluetoothAdapter;
12. import android.bluetooth.BluetoothDevice;
13. import android.bluetooth.BluetoothSocket;
14. import android.content.Context;
15. import android.hardware.Sensor;
16. import android.hardware.SensorEvent;
17. import android.hardware.SensorEventListener;
18. import android.hardware.SensorManager;
19. import android.os.Bundle;
20. import android.os.Handler;
21. import android.os.Message;
22. import android.os.Vibrator;
23. import android.util.Log;
24. import android.view.KeyEvent;
25. import android.widget.TextView;
26. import android.widget.Toast;
27. public class KoshiActivity extends Activity {
28.     private Sensor orientationSensor;
29.     private Sensor accelerometerSensor;
30.     private Sensor linearaccelerationSensor;
31.     private Sensor gyroscopeSensor;
32.     private Sensor rotationvectorSensor;
33.     private float[] orientationValues = new float[3];
34.     private float[] accelerometerValues = new float[3];
35.     private float[] linearaccelerationValues = new float[3];
36.     private float[] gyroscopeValues = new float[3];
37.     private float[] rotationvectorValues = new float[3];
38.     private String sendText;
39.     private String orientationText;
40.     private String accelerometerText;
41.     private String linearaccelerationText;
42.     private String gyroscopeText;
43.     private String rotationvectorText;
44.     SensorManager sm;
45.     private static final String TAG = "BLUETOOTH_SAMPLE";
46.     BluetoothAdapter mAdapter;
47.     Vector<BluetoothDevice> mDevices;
48.     TextView clientTextView;
49.     Context mContext;
50.     final String NAME = "BLUETOOTH_SAMPLE";
51.     UUID MY_UUID = UUID.fromString("11111111-0000-1000-1111-00AAECCAAFF");
52.     ClientThread clientThread;
53.     ConnectedThread connection;
54.     boolean send = false;
55.     boolean ss = false;
56.     boolean start = false;
57.     String values;
58.     Thread thread;
59.     TextView koshiText;
60.     private Runnable loop;
61.     private SensorEventListener listener = new SensorEventListener() {
62.     public void onAccuracyChanged(Sensor sensor, int accuracy) {
63.     }
64.     public void onSensorChanged(SensorEvent event) {
65.     if (event.sensor.getType() == Sensor.TYPE_LINEAR_ACCELERATION) {
66.     linearaccelerationValues = event.values;
67.     linearaccelerationText = String.valueOf(Math.round(Math.abs(linearaccelerationValues[0]) + Math.round(Math.abs(linearaccelerationValues[2])),1));
68.     }
69.     sendText = linearaccelerationText + "ita00";
70.     values = sendText;
71.     }
72.     };
73.     @Override
74.     public void onCreate(Bundle savedInstanceState) {
75.     super.onCreate(savedInstanceState);
76.     setContentView(R.layout.koshi);
77.     koshiText = (TextView) findViewById(R.id.koshiText);
78.     sm = (SensorManager) getSystemService(SENSOR_SERVICE);
79.     List<Sensor> sensorList;
80.     sensorList = sm.getSensorList(Sensor.TYPE_LINEAR_ACCELERATION);
81.     linearaccelerationSensor = sensorList.get(0);
82.     sm.registerListener(listener, linearaccelerationSensor,
83.     SensorManager.SENSOR_DELAY_GAME);
84.     mDevices = new Vector<BluetoothDevice>();
85.     mAdapter = BluetoothAdapter.getDefaultAdapter();
86.     Set<BluetoothDevice> devices = mAdapter.getBondedDevices();
87.     for (BluetoothDevice device : devices) {
88.     mDevices.add(device);
89.     Toast.makeText(this, "Name:" + device.getName(),
90.     Toast.LENGTH_SHORT).show();
91.     }
92.     for (int i = 0; i < mDevices.size(); i++) {
93.     clientThread = new ClientThread(mDevices.elementAt(i));
94.     clientThread.start();
95.     }
96.     loop = new Runnable() {
97.     public void run() {
98.     while (true) {
99.     try {
100.     Thread.sleep(50);
101.     } catch (InterruptedException e) {
102.     }
103.     if (start) {
104.     if (values != null && !values.equals("")) {
105.     connection.write(values.getBytes());
106.     }
107.     }
108.     }
109.     }
110.     };

```

```

111. }
112. private class ClientThread extends Thread
    {
113.     private final BluetoothSocket mmSocket;
114.     @SuppressWarnings("unused")
115.     private final BluetoothDevice mmDevice;
116.     public ClientThread(BluetoothDevice device)
    {
117.         BluetoothSocket tmp = null;
118.         mmDevice = device;
119.         try {
120.             tmp = device.createRfcommSocketToServiceRecord(MY_UUID);
121.         } catch (IOException e) {
122.             }
123.         mmSocket = tmp;
124.     }
125.     @Override
126.     public void run() {
127.         mAdapter.cancelDiscovery();
128.         try {
129.             mmSocket.connect();
130.         } catch (IOException connectException) {
131.             try {
132.                 mmSocket.close();
133.             } catch (IOException closeException) {
134.                 }
135.             return;
136.         }
137.         manageConnectedSocket(mmSocket);
138.         Message msg = new Message();
139.         msg.obj = "koshi";
140.         mHandler1.sendMessage(msg);
141.         start = true;
142.         thread = new Thread(looper);
143.         thread.start();
144.     }
145.     @SuppressWarnings("unused")
146.     public void cancel() {
147.         try {
148.             mmSocket.close();
149.         } catch (IOException e) {
150.             }
151.     }
152. }
153. public void manageConnectedSocket(BluetoothSocket socket) {
154.     connection = new ConnectedThread(socket);
155.     connection.start();
156. }
157. private class ConnectedThread extends Thread
    {
158.     private final BluetoothSocket mmSocket;
159.     private final InputStream mmInStream;
160.     private final OutputStream mmOutStream;
161.     String[] strAry = new String[5];
162.     long[] longAry = new long[5];
163.     public ConnectedThread(BluetoothSocket socket)
    {
164.         mmSocket = socket;
165.         InputStream tmpIn = null;
166.         OutputStream tmpOut = null;
167.         try {
168.             tmpIn = socket.getInputStream();
169.             tmpOut = socket.getOutputStream();
170.         } catch (IOException e) {
171.             }
172.         mmInStream = tmpIn;
173.         mmOutStream = tmpOut;
174.     }
175.     @Override
176.     public void run() {
177.         byte[] buffer = new byte[1024];
178.         @SuppressWarnings("unused")
179.         int bytes;
180.         while (true) {
181.             try {
182.                 bytes = mmInStream.read(buffer);
183.                 String readMsg = new String(buffer, "UTF-8");
184.                 Message msg = new Message();
185.                 msg.obj = readMsg;
186.                 strAry = readMsg.split("ita");
187.                 for(int i=0; i<longAry.length; i++){
188.                     longAry[i] = Long.parseLong(strAry[i]);
189.                 }
190.                 vibration(longAry);
191.                 mHandler.sendMessage(msg);
192.             } catch (IOException e) {
193.                 break;
194.             }
195.         }
196.     }
197.     public void write(byte[] bytes) {
198.         try {
199.             mmOutStream.write(bytes);
200.         } catch (IOException e) {
201.             }
202.     }
203.     @SuppressWarnings("unused")
204.     public void cancel() {
205.         try {
206.             mmSocket.close();
207.         } catch (IOException e) {
208.             }
209.     }
210.     public void vibration(long[] longAry){
211.         Vibrator vibrator = (Vibrator) getSystemService(Context.VIBRATOR_SERVICE);
212.         vibrator.vibrate(longAry, 1);
213.     }
214. }
215. final Handler mHandler = new Handler() {
216.     public void handleMessage(Message msg) {
217.     }
218. };
219. final Handler mHandler1 = new Handler() {
220.     public void handleMessage(Message msg) {
221.         koshiText.setText("腰付近のポケットに¥1000を入れてください");
222.     }
223. };
224. public float Marumeru(float f, int i) {
225.     BigDecimal xbi = new BigDecimal(String.valueOf(f));
226.     float result = xbi.setScale(i, BigDecimal.ROUND_HALF_UP).floatValue();
227.     return result;
228. }
229. public boolean onKeyDown(int keyCode, KeyEvent event) {
230.     if (keyCode == KeyEvent.KEYCODE_BACK) {
231.         System.exit(0);
232.     } else if (keyCode == KeyEvent.KEYCODE_MENU) {
233.         openOptionsMenu();
234.     }
235.     return true;
236. }
237. }

```

2-4 教師用システム（グラフ表示のプログラム）

```

1. <?php
2. date_default_timezone_set ('Asia/Tokyo');
3. session_start ();
4. if (! isset ( $_SESSION ["user_name"] ))
5. {
6. $no_login_url = "../login.php";
7. header ( "Location: {$no_login_url}");
8. exit ();
9. }
10. function checkSelected($var, $selected_value) {
11. if ($var == $selected_value) {
12. return ' selected="selected"';
13. }
14. return '';
15. }
16. $sort_ary = array (
17. "5" => "日時降順",
18. "1" => "速度が速い順",
19. "2" => "速度が遅い順",
20. "3" => "ブレ値が大きい順",
21. "4" => "ブレ値が小さい順"
22. );
23. function moveAverage($data) {
24. $sum = 0;
25. $n = 0;
26. for($i = 0; $i < count ( $data ); $i++) {
27. $sum = ($sum) + ($data [$i]);
28. $n ++;
29. }
30. $result = ($sum) / ($n);
31. return $result;
32. }
33. function mathGraphData($userData, $gap) {
34. $var = array ();
35. $var = explode ( ",", $userData );
36. for($i = $gap; $i < count ( $var ) + $gap;
37. $i = $i + 5) {
38. $returnData = $returnData . "," . (($var
39. [($i - 2)] + $var [($i - 1)] + $var [$i] +
40. $var [($i + 1)] + $var [($i + 2)]) / 5);
41. }
42. return $returnData;
43. }
44. function mathGraphGyroData($userData, $gap) {
45. $var = array ();
46. $var = explode ( ",", $userData );
47. for($i = $gap; $i < count ( $var ) + $gap;
48. $i = $i + 5) {
49. $returnData = $returnData . "," . ($var
50. [$i] * 2);
51. }
52. return $returnData;
53. }
54. $class_list = $_GET ['class_list'];
55. $kikan = $_GET ['kikan'];
56. $sort_num = $_GET ['sort_num'];
57. $search_name = $_GET ['search_name'];
58. $view = $_GET ['view'];
59. $p = $_GET ['p'];
60. $time = time ();
61. $ID_ar = array ();
62. $uTime_ar = array ();
63. $speed_ar = array ();
64. $acc_ar = array ();
65. $gyro_ar = array ();
66. $koshi_ar = array ();
67. $gap_ar = array ();
68. $accPoint_ar = array ();
69. $gyroPoint_ar = array ();
70. $koshiPoint_ar = array ();
71. $totalPoint_ar = array ();
72. $year_ar = array ();
73. $class_ar = array ();
74. $number_ar = array ();
75. $sex_ar = array ();
76. $name_ar = array ();
77. $modelID = "";
78. $modelSpeed = "";
79. $modelAcc = "";
80. $modelGyro = "";
81. $i = 0;
82. $n = 0;
83. try {
84. $pdo = new PDO ("sqlite:../elearning.db"
85. );
86. $pdo->setAttribute (PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
87. chmod ( "elearning.db", 0777 );
88. if (! $pdo) {
89. die ( '接続失敗です。 ' . $sqliteerror );
90. } else {
91. $isUserTb = false;
92. $isDatatb = false;
93. $isTypetb = false;
94. print ('接続に成功しました。 <br>');
95. }
96. $sql_query1 = "SELECT plainTb.*, userTb.y
97. ear, userTb.class, userTb.number,
98. userTb.sex, nameTb.name FROM plainTb, use
99. rTb, nameTb ON
100. plainTb.ID=userTb.ID and plainTb.ID=nameT
101. b.ID";
102. $sql_query2 = "SELECT count (*) FROM plain
103. Tb , userTb ON userTb.ID=plainTb.ID";
104. if ($kikan != null && $kikan != "null"){
105. if ($kikan == 1) {
106. $viewtime = $time - 86400;
107. $sql_query = $sql_query . " where uTime >
108. $viewtime";
109. } else if ($kikan == 2) {
110. $viewtime = $time - 604800;
111. $sql_query = $sql_query . " where uTime >
112. $viewtime";
113. } else if ($kikan == 3) {
114. $viewtime = $time - 2419200;
115. $sql_query = $sql_query . " where uTime >
116. $viewtime";
117. }
118. }
119. if ($search_name != "") {
120. foreach ( $pdo->query (
121. "SELECT * FROM nameTb WHERE name LIKE '%
122. $search_name%'") as $row){
123. $viewuserid = $row ["ID"];
124. }
125. $sql_query = $sql_query . " and userTb.ID
126. = ¥"$viewuserid¥";
127. }
128. if (0 < $class_list && $class_list < 5) {
129. $sql_query = $sql_query . " and class = ¥"
130. $class_list¥";
131. }
132. if (0 < $sort_num && $sort_num < 6) {
133. if ($sort_num == 1) {
134. $sql_query = $sql_query . " ORDER BY accP
135. oint DESC";
136. } else if ($sort_num == 2) {
137. $sql_query = $sql_query . " ORDER BY accP

```



```

241. foreach ( $pdo->query ( "SELECT modelID F
ROM plainSetTb" ) as $row ) {
242. $modelID = $row ["modelID"];
243. }
244. foreach ( $pdo->query ( "SELECT * FROM pl
ainModelTb WHERE modelID = '$modelID'" )
as $row ) {
245. $modelSpeed = $row ["speed"];
246. $modelAcc = $row ["acc"];
247. $modelGyro = $row ["gyro"];
248. $modelGap = $row ["gap"];
249. }
250. }
251. foreach ( $pdo->query ( $sql_query1 ) as
$row ) {
252. array_push ( $ID_ar, $row ["ID"] );
253. array_push ( $uTime_ar, $row ["uTime"] );
254. array_push ( $speed_ar, $row ["speed"] );
255. array_push ( $acc_ar, $row ["acc"] );
256. array_push ( $gyro_ar, $row ["gyro"] );
257. array_push ( $koshi_ar, $row ["koshi"] );
258. array_push ( $gap_ar, $row ["gap"] );
259. array_push ( $accPoint_ar, $row ["accPoin
t"] );
260. array_push ( $gyroPoint_ar, $row ["gyroPo
int"] );
261. array_push ( $koshiPoint_ar, $row ["koshi
Point"] );
262. array_push ( $totalPoint_ar, $row ["total
Point"] );
263. array_push ( $year_ar, $row ["year"] );
264. array_push ( $class_ar, $row ["class"] );
265. array_push ( $number_ar, $row ["number"]
);
266. array_push ( $sex_ar, $row ["sex"] );
267. array_push ( $name_ar, $row ["name"] );
268. $n ++;
269. }
270. $rader = "";
271. $line = array ();
272. $line2 = array ();
273. $gyr = array ();
274. for($j = 0; $j < $n; $j ++ ) {
275. $datastr = "";
276. $databurestr = "";
277. $data_gyro = explode ( ",", $gyro_ar [$j]
);
278. for($count = 0; $count < count ( $data_gy
ro ) - 1; $count ++ ) {
279. $databurestr = $databurestr . (($data_gyr
o [$count - 1] + $data_gyro [$count] + $da
ta_gyro [$count + 1]) * 100 / 3) . ",";
280. }
281. $databurestr = $databurestr . abs ( $data
_gyro [count ( $data_gyro ) - 1] *
282. 100 );
283. $gap = $gap_ar [$j] - $modelGap;
284. $graphData = mathGraphData ( $speed_ar
[$j], $gap );
285. $graphDataM = mathGraphData ( $modelSpeed,
0 );
286. $graphDataA = mathGraphData ( $acc_ar
[$j], $gap );
287. $graphDataAM = mathGraphData ( $modelAcc,
0 );
288. $databurestr = mathGraphGyroData ( $datab
urestr, $gap );
289. $line [] = "
290. var lineChartData$j = {
291. labels : [$yAxis],
292. datasets : [{
293. fillColor : 'rgba(99, 37, 35, 0)',
294. strokeColor : 'rgba(99, 37, 35, 1)',
295. pointDot : false,
296. data : [$graphData]
297. }, {
298. fillColor : 'rgba(99, 37, 35, 0)',
299. strokeColor : 'rgba(82,154,157, 1)',
300. pointDot : false,
301. data : [$graphDataM]
302. }, {
303. fillColor : 'rgba(207,153,59,0.5)',
304. strokeColor : 'rgba(207,153,59,1)',
305. pointDot : false,
306. data : [$databurestr]
307. } ]
308. }
309. var option$j = {
310. scaleSteps : 3,
311. scaleStepWidth : 1000,
312. scaleStartValue : 0
313. }
314. var myLine$j = new Chart(document.getElem
entById('canvas$j')
315. .getContext('2d')).Line(lineChartData$j,
option$j);";
316. $line2 [] = "
317. var lineChartData2$j = {
318. labels : [$yAxis],
319. datasets : [{
320. fillColor : 'rgba(99, 37, 35, 0)',
321. strokeColor : 'rgba(82,154,157, 1)',
322. pointDot : false,
323. data : [$graphDataAM]
324. }, {
325. fillColor : 'rgba(99, 37, 35, 0)',
326. strokeColor : 'rgba(99, 37, 35, 1)',
327. pointDot : false,
328. data : [$graphDataA]
329. } ]
330. }
331. var option2$j = {
332. scaleSteps : 4, scaleStepWidth : 20,
333. scaleStartValue : -40,
334. scaleLabel : ¥" <%=value%>¥"
335. }
336. var myLine2$j = new Chart(document.getEle
mentById('canvas2$j')
337. .getContext('2d')).Line(lineChartData2$j,
option2$j);";
338. $rader_data = array (
339. $accPoint_ar [$j],
340. $gyroPoint_ar [$j],
341. $koshiPoint_ar [$j]
342. );
343. $rader .= "
344. var rc$j = new html5jp.graph.radar('rader
$j');
345. if( ! rc$j ) { return; }
346. var items = [
347. ['xxx',$rader_data[0],$rader_data[1],$rad
er_data[2]]
348. ];
349. var params = {
350. aCap: ['速さ', 'ブレ', '腰'],
351. aMax: 100,
352. aMin: 0,
353. chartShape: 'circle',
354. aLinePositions: [0,20,40,60,80,100]
355. };
356. rc$j.draw(items, params);";

```

```

357. echo "<script type='text/javascript' src='
'rendering-mode.js'></script>
358. <!--[if IE]><script type='text/javascript
' src='html5jp/
359. excanvas/excanvas.js'></script><![endif]-
->
360. <script type='text/javascript' src='html5
jp/graph/radar.js'></script>
361. <script type='text/javascript'>
362. window.onload = function() {
363. $rader
364. };
365. </script>;
366. $date = date ( 'm/d H:i', $uTime_ar [$i]
);
367. echo "
368. <div id='userData'>
369. <table align='center' valign='middle' bo
rder='0'>
370. <TR>
371. <TD>
372. <table class='sample' >
373. <TR>
374. <TH>名前</TH>
375. <TD><div class='buttonDiv'><A HREF = grow
thRecord.php?
376. ID={$ID_ar[$i]}>$name_ar[$i]</A></div></
TD>
377. </TR>
378. <TR>
379. <TH>番号</TH>
380. <TD>${class_ar[$i]}組$number_ar[$i]番</TD>
381. </TR>
382. <TR>
383. <TH>練習日時</TH>
384. <TD>$date</TD>
385. </TR>
386. <TR>
387. <TH>引きの速度</TH>
388. <TD>${accPoint_ar[$i]}点</TD>
389. </TR>
390. <TR>
391. <TH>引きのブレ</TH>
392. <TD>${gyroPoint_ar[$i]}点</TD>
393. </TR>
394. <TR>
395. <TH>腰の使い方</TH>
396. <TD>${koshiPoint_ar[$i]}点</TD>
397. </TR>
398. <TR>
399. <TH>総合得点</TH>
400. <TD>${totalPoint_ar[$i]}点</TD>
401. </TR>
402. </TABLE>
403. </TD>
404. <TD>
405. <div class=zoomTarget>
406. <BR>
407. <canvas id='canvas$i' height='140' width=
'400'></canvas>
408. <script> $line[$i] </script>
409. </div>
410. <div class=zoomTarget>
411. <BR>
412. <canvas id='canvas2$i' height='140' width
='400'></canvas>
413. <script> $line2[$i] </script>
414. </div>
415. </TD>
416. <TD>
417. <div class=zoomTarget>
418. <div><canvas width='300' height='300' id=
'rader$i'></div>
419. </div>
420. </TD>
421. </TR>
422. </table>
423. </div>;
424. $i ++;
425. $datastr = "";
426. $databurestr = "";
427. }
428. } catch ( PDOException $e ) {
429. print $e->getMessage ();
430. die ();
431. }
432. $pdo = null;
433. echo "<table align='center' valign='midd
le'><tr>";
434. $urlname = "&class_list=${class_list}&sort_
num=${sort_num}
435. &kikan=${kikan}&search_name=${search_name}&vi
ew=${view}";
436. if ($p > 0) {
437. $Prev = $p - 1;
438. $PrevPage = "<td><div class='buttonDiv'><
a href='graphview.php
439. ?p=${Prev}${urlname}'>前の10件</a></div><
/td>";
440. echo $TopPage = "<td><div class='buttonDi
v'><a href='graphview.php
441. ?${urlname}'><</a></div></td> ";
442. echo $PrevPage . " ";
443. }
444. if ($p < $maxpage) {
445. $Next = $p + 1;
446. $NextPage = "<td><div class='buttonDiv'><
a href='graphview.php
447. ?p=${Next}${urlname}'>次の10件</a></div><
/td> ";
448. echo $NextPage;
449. echo $TopPage = "<td><div class='buttonDi
v'><a href='graphview.php
450. ?${urlname}&p=${maxpage}'>>></a></div></t
d>";
451. }
452. echo "</tr></table><BR>";
453. ?>
454. <div class='buttonDiv' align="right">
455. <a href="#maindiv">▲ページの先頭へ戻る</a>
456. </div>
457. </div>
458. <BR>
459. </body>
460. </html>

```

2-5 教師用システム（リスト表示のプログラム）

```

1. <?php
2. date_default_timezone_set('Asia/Tokyo');
3. session_start();
4. if (!isset($_SESSION["user_name"])) {
5. $no_login_url = "../login.php";
6. header("Location: {$no_login_url}");
7. exit;
8. }
9. function viewData($miss, $good) {
10. if ($miss > 1) {

```

```

11. return "bgcolor='feldsper'";
12. }else if($good==3){
13. return "bgcolor='LightCyan'";
14. }
15. return '';
16. }
17. function getSortId($var){
18. if($var!=null){
19. if($var==1||$var==3){
20. $var=1;
21. }else{
22. $var=2;
23. }
24. return $var;
25. }
26. }
27. ini_set('display_errors', 0);
28. $ID_ar = array();
29. $class_ar = array();
30. $number_ar = array();
31. $name_ar = array();
32. $totalPoint_ar = array();
33. $speed_ar = array();
34. $gyro_ar = array();
35. $koshi_ar = array();
36. $uTime_ar = array();
37. $i=0;
38. $sortNumber = $_GET['sortNumber'];
39. $sortNumber = getSortId($sortNumber);
40. $sortPoint = $_GET['sortPoint'];
41. $sortPoint = getSortId($sortPoint);
42. $sortSpeed = $_GET['sortSpeed'];
43. $sortSpeed = getSortId($sortSpeed);
44. $sortBure = $_GET['sortBure'];
45. $sortBure = getSortId($sortBure);
46. $sortKoshi = $_GET['sortKoshi'];
47. $sortKoshi = getSortId($sortKoshi);
48. $sortDate = $_GET['sortDate'];
49. $sortDate = getSortId($sortDate);
50. if($_POST["selectClass"]!=null){
51. $_SESSION["selectClass"] = $_POST["select
Class"];
52. }
53. ?>
54. <html>
55. <head>
56. <title>一覧画面</title>
57. <link rel="stylesheet" href="list.css" ty
pe="text/css" />
58. </head>
59. <body>
60. <div class="mainviv">
61. <div class='buttonDiv'><a href="..">PLMS
トップへ</a></div>
62. <br />
63. <div class='buttonDiv'><a href="graphview.
php">グラフ表示へ切り替え</a></div><br>
64. クラス選択 :
65. <form style="display:inline;" action="lis
tview.php" method="post">
66. <select name='selectClass' onChange='this
form.submit()'>
67. <option value='0'>すべて</option>
68. <option value='1' <?php if($_SESSION["sel
ectClass"] == 1){echo "selected";}?>>1組<
/option>
69. <option value='2' <?php if($_SESSION["sel
ectClass"] == 2){echo "selected";}?>>2組<
/option>
70. <option value='3' <?php if($_SESSION["sel
ectClass"] == 3){echo "selected";}?>>3組<
/option>
71. <option value='4' <?php if($_SESSION["sel
ectClass"] == 4){echo "selected";}?>>4組<
/option>
72. </select>
73. </form>
74. <hr width="1000" size="2" align="left">
75. <?php
76. try{
77. $pdo = new PDO("sqlite://elearning.db");
78. $pdo->setAttribute(PDO::ATTR_ERRMODE, PD
O::ERRMODE_EXCEPTION);
79. $sql_query = "SELECT plainTb.*, userTb.ye
ar, userTb.class, userTb.number, userTb.s
ex, nameTb.name FROM plainTb, userTb, nam
eTb ON plainTb.ID=userTb.ID and plainTb.I
D=nameTb.ID ";
80. if($_SESSION["selectClass"]>0){
81. $sql_query = $sql_query." AND class = ".
$_SESSION['selectClass'];
82. }
83. if($sortNumber==1){
84. $sql_query = $sql_query." ORDER BY number
ASC";
85. }else if($sortNumber==2){
86. $sql_query = $sql_query." ORDER BY number
DESC";
87. }else if($sortPoint==1){
88. $sql_query = $sql_query." ORDER BY totalP
oint DESC";
89. }else if($sortPoint==2){
90. $sql_query = $sql_query." ORDER BY totalP
oint ASC";
91. }else if($sortSpeed==1){
92. $sql_query = $sql_query." ORDER BY accPoin
t ASC";
93. }else if($sortSpeed==2){
94. $sql_query = $sql_query." ORDER BY accPoin
t DESC";
95. }else if($sortBure==1){
96. $sql_query = $sql_query." ORDER BY gyroPoi
nt DESC";
97. }else if($sortBure==2){
98. $sql_query = $sql_query." ORDER BY gyroPoi
nt ASC";
99. }else if($sortKoshi==1){
100. $sql_query = $sql_query." ORDER BY koshiP
oint DESC";
101. }else if($sortKoshi==2){
102. $sql_query = $sql_query." ORDER BY koshiP
oint ASC";
103. }else if($sortDate == 1){
104. $sql_query = $sql_query." ORDER BY uTime
ASC";
105. }else if($sortDate == 2){
106. $sql_query = $sql_query." ORDER BY uTime
DESC";
107. }else{
108. $sql_query = $sql_query." ORDER BY uTime
DESC";
109. }
110. $sortNumber++;
111. $sortPoint++;
112. $sortSpeed++;
113. $sortBure++;
114. $sortKoshi++;
115. $sortDate++;
116. ?>
117. <table align='center' valign='middle' cla
ss='sample'>
118. <tr align = 'center'><th>クラス</th>
119. <?php echo "
120. <th><div class='buttonDiv'><a href = 'lis
tview.php?sortNumber=$sortNumber'>番号</a
></div></th>

```

```

121. <TH>名前</TH>
122. <TH><div class='buttonDiv'><A HREF = 'listview.php?sortPoint=$sortPoint'>得点</a></div></TH>
123. <TH><div class='buttonDiv'><A HREF = 'listview.php?sortSpeed=$sortSpeed'>引く速さ</a></div></TH>
124. <TH><div class='buttonDiv'><A HREF = 'listview.php?sortBure=$sortBure'>プレの少なさ</a></div></TH>
125. <TH><div class='buttonDiv'><A HREF = 'listview.php?sortKoshi=$sortKoshi'>腰の使い方</a></div></TH>
126. <TH><div class='buttonDiv'><A HREF = 'listview.php?sortDate=$sortDate'>日時</a></div></TH>";
127. foreach($pdo->query($sql_query) as $row){
128. array_push($ID_ar,$row['ID']);
129. array_push($class_ar,$row['class']);
130. array_push($number_ar,$row['number']);
131. array_push($name_ar,$row['name']);
132. array_push($totalPoint_ar,(int)$row['totalPoint']);
133. array_push($speed_ar,$row['accPoint']);
134. array_push($gyro_ar,$row['gyroPoint']);
135. array_push($koshi_ar,$row['koshiPoint']);
136. array_push($uTime_ar,$row['uTime']);
137. $missInt = 0;
138. $goodInt = 0;
139. if ($totalPoint_ar[$i] < 50) {
140. $totalPoint_ar[$i] = "<font color='red'>$totalPoint_ar[$i]点</font>";
141. } else {
142. $totalPoint_ar[$i] = "$totalPoint_ar[$i]点";
143. }
144. if($speed_ar[$i]<60){
145. $speed_ar[$i] = "<font color='red'>★</font>";
146. $missInt ++;
147. }else if($speed_ar[$i]<=80){
148. $speed_ar[$i] = "★★";
149. }else{
150. $speed_ar[$i] = "★★★";
151. $goodInt ++;
152. }
153. if($gyro_ar[$i]<60){
154. $gyro_ar[$i] = "<font color='red'>★</font>";
155. $missInt ++;
156. }else if($gyro_ar[$i]<=80){
157. $gyro_ar[$i] = "★★";
158. }else{
159. $gyro_ar[$i] = "★★★";
160. $goodInt ++;
161. }
162. if($koshi_ar[$i]<60){
163. $koshi_ar[$i] = "<font color='red'>★</font>";
164. $missInt ++;
165. }else if($koshi_ar[$i]<=80){
166. $koshi_ar[$i] = "★★";
167. }else{
168. $koshi_ar[$i] = "★★★";
169. $goodInt ++;
170. }
171. $timeview = date("m/d H:i",$uTime_ar[$i]);
172. $viewColor = viewData($missInt,$goodInt);
173. echo "
174. <TR ALIGN = 'center'><TD>$class_ar[$i]</TD>
175. <TD>$number_ar[$i]</TD>
176. <TD $viewColor><A HREF = 'growthRecord.php?ID=$ID_ar[$i]'>$name_ar[$i]</A></TD>
177. <TD>$totalPoint_ar[$i]</TD>
178. <TD Align='left'>$speed_ar[$i]</TD>
179. <TD Align='left'>$gyro_ar[$i]</TD>
180. <TD Align='left'>$koshi_ar[$i]</TD>
181. <TD>$timeview</TD>
182. </TR>";
183. $i++;
184. }
185. }catch(PDOException $e){
186. print $e->getMessage();
187. die();
188. }
189. echo "</table>";
190. $pdo = null;
191. ?>
192. </div>
193. <BR>
194. <div class='buttonDiv' align="right">
195. <a href="#maindiv">▲ページの先頭へ戻る</a>
196. </div>
197. </body>
198. </html>

```

2-6 教師用システム（上達履歴表示のプログラム）

```

1. <?php
2. session_start();
3. if (!isset($_SESSION["user_name"])) {
4. $no_login_url = "../login.php";
5. header("Location: {$no_login_url}");
6. exit;
7. } else {
8. }
9. $ID = $_GET['ID'];
10. $writeCsvDecision = $_GET['write'];
11. function connectionStr($var,$useData,$totalData){
12. if($totalData==0){
13. $var = round($useData);
14. }else{
15. $var = $var.".".round($useData);
16. }
17. return $var;
18. }
19. function graphWrite($xData,$yData,$num){
20. $var="
21. <script>
22. var lineChartData$num = {
23. labels : [$yData],
24. datasets : [
25. {
26. fillColor : 'rgba(99, 37, 35, 0)',
27. strokeColor : 'rgba(99, 37, 35, 1)',
28. pointDot : false,
29. data : [$xData]
30. }
31. ]
32. }
33. var myLine$num = new Chart(document.getElementById('canvas$num')).getContext('2d').Line(lineChartData$num);
34. </script>
35. ";
36. return $var;
37. }
38. try {
39. $pdo = new PDO("sqlite:../elearning.db");

```

```

40. $pdo -> setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
41. if (!$pdo) {
42. die('接続失敗です。' . $sqliteerror);
43. } else {
44. }
45. }catch(PDOException $e) {
46. print $e -> getMessage();
47. die();
48. }
49. $sql_query = "SELECT plainTb.*, nameTb.name FROM plainTb, nameTb ON plainTb.ID=nameTb.ID WHERE plainTb.ID = $ID ORDER BY updateTime ASC";
50. $totalPoint = "";
51. $accPoint = "";
52. $gyroPoint = "";
53. $koshiPoint = "";
54. $totalData=0;
55. foreach ($pdo->query($sql_query) as $row) {
56. $userName = $row["name"];
57. $totalPoint .= $row["totalPoint"].",";
58. $accPoint .= $row["accPoint"].",";
59. $gyroPoint .= $row["gyroPoint"].",";
60. $koshiPoint .= $row["koshiPoint"].",";
61. $totalData++;
62. }
63. $totalDataHalf = round($totalData / 2);
64. ?>
65. <html>
66. <head>
67. <title><?php echo $userName?>さんの記録</title>
68. <link rel="stylesheet" href="growth.css" type="text/css" />
69. <script
70. src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js"></script>
71. <script type="text/javascript" src="./js/jquery.zoomooz.min.js">
72. </script>
73. <script type="text/javascript">
74. $(document).ready(function() {
75. $("zoomTarget").zoomTarget();
76. });
77. </script>
78. <script type="text/javascript" src="./js/parts.js"></script>
79. <script src="ChartGr.js"></script>
80. <meta name = "viewport" content = "initial-scale = 1, user-scalable = no">
81. <style>
82. canvas{
83. }
84. </style>
85. </head>
86. <body align="center">
87. <div class='buttonDiv'>
88. <a href="..">PLMSトップ</a>
89. <a href="graphview.php">グラフ表示</a>
90. <a href="listview.php">リスト表示</a>
91. </div><br /><br />
92. <div class='buttonDiv'><a href = 'graphview.php?search_name=<?php echo ("userName"); ?>'><?php echo ("userName"); ?>さんの1回ごとの結果</a></div><br>
93. <div id='growTop'>
94. <p>
95. <font size="6" color="#ffffff"><?php echo $userName?>さんの記録</font>
96. </p>
97. </div>
98. <div id="main" align="center">
99. <?php
100. $x = "";
101. for($j=1; $j<$totalData; $j++) {
102. $x .= $j.',';
103. if ($j == $totalData-1) {
104. $x .= $totalData;
105. }
106. }
107. echo ("<table align = 'center'><tr align = 'center'><td>");
108. echo "総合得点<br /><canvas id='canvas0' height='400' width='500'></canvas><br /><br />";
109. echo graphWrite($totalPoint,$x,0);
110. echo ("</td>");
111. echo ("<td>");
112. echo "ひきの速さ<br /><canvas id='canvas1' height='400' width='500'></canvas><br /><br />";
113. echo graphWrite($accPoint,$x,1);
114. echo ("</td></tr>");
115. echo ("<tr align = 'center'><td>");
116. echo "ブレの少なさ<br /><canvas id='canvas2' height='400' width='500'></canvas><br /><br />";
117. echo graphWrite($gyroPoint,$x,2);
118. echo ("</td>");
119. echo ("<td>");
120. echo "腰の使い方<br /><canvas id='canvas3' height='400' width='500'></canvas><br /><br />";
121. echo graphWrite($koshiPoint,$x,3);
122. echo ("</td></tr></table>");
123. ?>
124. </div>
125. <BR><BR>
126. <div class='buttonDiv'><a href = 'graphview.php?search_name=<?php echo ("userName"); ?>'><?php echo ("userName"); ?>さんの1回ごとの結果</a></div>
127. <div class='buttonDiv'>
128. <br /><br />
129. <div class='buttonDiv'>
130. <a href="..">PLMSトップ</a>
131. <a href="graphview.php">グラフ表示</a>
132. <a href="listview.php">リスト表示</a>
133. </div>
134. </body>
135. </html>

```

2-7 教師用システム（学習者用アプリケーションからのデータを受け取るプログラム）

```

1. <html>
2. <body>
3. <?php
4. date_default_timezone_set('Asia/Tokyo');
5. function calcPoint($var){
6. if($var>100){
7. $var=100;
8. }else if($var<0){
9. $var=0;
10. }
11. return $var;
12. }

```

```

13. function median($list){
14. sort($list);
15. if (count($list) % 2 == 0){
16. return (($list[(count($list)/2)-1]+$list
17. [(count($list)/2)])/2);
18. }else{
19. return ($list[floor(count($list)/2)]);
20. }
21. function average($list){
22. $sum = 0;
23. for ($i=0; $i<count($list); $i++) {
24. $sum = $sum + $list[$i];
25. }
26. return $sum/count($list);
27. }
28. $ID = $_GET['ID'];
29. $speed = $_GET['speed'];
30. $acc = $_GET['acc'];
31. $gyro = $_GET['gyro'];
32. $koshi = $_GET['koshi'];
33. $gap = $_GET['gap'];
34. $accPoint = $_GET['accPoint'];
35. $gyroPoint = $_GET['gyroPoint'];
36. $koshiPoint = $_GET['koshiPoint'];
37. $totalPoint = $_GET['totalPoint'];
38. $uTime = time();
39. $speedArray = explode(",", $speed);
40. $accArray = explode(",", $acc);
41. $gyroArray = explode(",", $gyro);
42. $koshiArray = explode(",", $koshi);
43. $num = count($accArray);
44. $acc = "";
45. for ($i=0; $i<$num; $i++) {
46. if ($i < 4 || $i > $num-5) {
47. $acc .= $accArray[$i].',';
48. if ($i == $num) {
49. $acc .= $accArray[$i];
50. }
51. } else {
52. $tempArray = array($accArray[$i-2], $accA
53. 1], $accArray[$i+2]);
54. $acc .= median($tempArray).',';
55. }
56. }
57. $num = count($gyroArray);
58. $gyro = "";
59. for ($i=0; $i<$num; $i++) {
60. if ($i < 4 || $i > $num-5) {
61. $gyro .= $gyroArray[$i].',';
62. if ($i == $num) {
63. $gyro .= $gyroArray[$i];
64. }
65. } else {
66. $tempArray = array($gyroArray[$i-2], $gyr
67. oArray[$i-1], $gyroArray[$i], $gyroArray
68. [$i+1], $gyroArray[$i+2]);
69. $gyro .= average($tempArray).',';
70. }
71. }
72. try{
73. $pdo = new PDO("sqlite:../elearning.db");
74. $pdo->setAttribute(PDO::ATTR_ERRMODE, PD
75. O::ERRMODE_EXCEPTION);
76. if (!$pdo) {
77. die('接続失敗です. '.$sqliteerror);
78. }else{
79. print('接続に成功しました. <br>');
80. }
81. $pdo->exec("INSERT INTO plainTb VALUES
82. ($ID,$uTime,'$speed','$acc','$gyro','$kos
83. hi','$gap',$accPoint,$gyroPoint,$koshiPoint,
84. $totalPoint);");
85. }catch(PDOException $e){
86. print $e->getMessage();
87. die();
88. }
89. $pdo = null;
90. print('切断しました. <br>');
91. ?>
92. </body>
93. </html>

```

2-8 Moodle プラグイン（データベースに作成するテーブルの情報を記すファイル）

```

1. <?xml version="1.0" encoding="UTF-8" ?>
2. <XMLDB PATH="question/type/motorskill/db"
   VERSION="20120122" COMMENT="XMLDB file for Moodle question/type/motorskill"
3. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4. xsi:noNamespaceSchemaLocation="../../../../lib/xmldb/xmldb.xsd">
5. <TABLES>
6. <TABLE NAME="question_motorskill" COMMENT="Options for motorskill questions.">
7. <FIELDS>
8. <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
9. <FIELD NAME="question" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="Redundant, because of the answer field. Foreign key references question.id."/>
10. <FIELD NAME="answer" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="Foreign key references question.answers.id."/>
11. <FIELD NAME="tolerance" TYPE="char" LENGTH="255" NOTNULL="true" DEFAULT="0.0" SEQUENCE="false" COMMENT="Allowed error when matching a response to this answer. I don't know why this is stored as a string."/>
12. </FIELDS>
13. <KEYS>
14. <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
15. <KEY NAME="question" TYPE="foreign" FIELDS="question" REFTABLE="question" REFFIELDS="id"/>
16. </KEYS>
17. <INDEXES>
18. <INDEX NAME="answer" UNIQUE="false" FIELDS="answer"/>
19. </INDEXES>
20. </TABLE>
21. <TABLE NAME="question_motorskill_options" COMMENT="Options for questions of type motorskill This table is also used by the calculated question type">
22. <FIELDS>
23. <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
24. <FIELD NAME="question" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false"/>
25. <FIELD NAME="showunits" TYPE="int" LENGTH="4" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="How units are handled: 3) Not used at all, 0) Optional, or 1) must be right or penalty applied."/>
26. <FIELD NAME="unitsleft" TYPE="int" LENGTH="4" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="display the unit at left as in $1.00"/>
27. <FIELD NAME="unitgradingtype" TYPE="int" LENGTH="4" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="0 no penalty, 1 fraction response grade, 2 fraction total grade"/>
28. <FIELD NAME="unitpenalty" TYPE="number" LENGTH="12" NOTNULL="true" DEFAULT="0.1" SEQUENCE="false" DECIMALS="7" COMMENT="Penalty for getting the unit wrong, when they are being graded."/>
29. </FIELDS>
30. <KEYS>
31. <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
32. <KEY NAME="question" TYPE="foreign" FIELDS="question" REFTABLE="question" REFFIELDS="id"/>
33. </KEYS>
34. </TABLE>
35. <TABLE NAME="question_motorskill_units" COMMENT="Optional unit options for motorskill questions. This table is also used by the calculated question type.">
36. <FIELDS>
37. <FIELD NAME="id" TYPE="int" LENGTH="10" NOTNULL="true" SEQUENCE="true"/>
38. <FIELD NAME="question" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false" COMMENT="Foreign key references question.id."/>
39. <FIELD NAME="multiplier" TYPE="number" LENGTH="10" NOTNULL="true" DEFAULT="1.0000" SEQUENCE="false" DECIMALS="5" COMMENT="The multiplier for this unit. For example, if the first unit is (1.0, 'cm'), another unit might be (0.1, 'mm') or (100.0, 'm')."/>
40. <FIELD NAME="unit" TYPE="char" LENGTH="50" NOTNULL="true" SEQUENCE="false" COMMENT="The unit. For example 'm' or 'kg'."/>
41. </FIELDS>
42. <KEYS>
43. <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
44. <KEY NAME="question" TYPE="foreign" FIELDS="question" REFTABLE="question" REFFIELDS="id"/>
45. </KEYS>
46. <INDEXES>
47. <INDEX NAME="question-unit" UNIQUE="true" FIELDS="question, unit" COMMENT="Unique index to ensure that only one unit with a particular name is created for each question."/>
48. </INDEXES>
49. </TABLE>
50. <TABLE NAME="question_motorskill_app" COMMENT="App ID for practice of motorskill">
51. <FIELDS>
52. <FIELD NAME="question" TYPE="int" LENGTH="10" NOTNULL="true" DEFAULT="0" SEQUENCE="false"/>
53. <FIELD NAME="app_id" TYPE="char" LENGTH="255" NOTNULL="true" DEFAULT="app" SEQUENCE="false"/>
54. </FIELDS>
55. <KEYS>
56. <KEY NAME="primary" TYPE="primary" FIELDS="question"/>
57. </KEYS>
58. </TABLE>
59. </TABLES>
60. </XMLDB>

```

2-9 Moodle プラグイン（問題の作成や編集を行うプログラム）

```

1. <?php
2.     defined('MOODLE_INTERNAL') || die();

```

```

3.   require_once($CFG->dirroot . '/question/t
type/edit_question_form.php');
4.   require_once($CFG->dirroot . '/question/t
type/motorskill/questiontype.php');
5.   class qtype_motorskill_edit_form extends
question_edit_form {
6.     const UNITS_MIN_REPEATS = 1;
7.     const UNITS_TO_ADD = 2;
8.     protected $ap = null;
9.     protected function definition_inner($mform) {
10.    $this->add_app_id($mform);
11.    $this->add_per_answer_fields($mform, get_
string('answer_no', 'qtype_motorskill', '
{no}'),
12.    question_bank::fraction_options());
13.    $this->add_unit_options($mform);
14.    $this->add_unit_fields($mform);
15.    $this->add_interactive_settings();
16.    }
17.    protected function get_per_answer_fields
($mform, $label, $gradeoptions, &$repeate
doptions, &$answersoption) {
18.    $repeated = parent::get_per_answer_fields
($mform, $label, $gradeoptions, $repeated
options, $answersoption);
19.    $tolerance = $mform->createElement('text',
'tolerance', get_string('answererror', '
qtype_motorskill'), array('size' => 15));
20.    $repeatedoptions['tolerance']['type'] = P
ARAM_FLOAT;
21.    $repeatedoptions['tolerance']['default']
= 0;
22.    $elements = $repeated[0]->getElements();
23.    $elements[0]->setSize(15);
24.    array_splice($elements, 1, 0, array($tol
erance));
25.    $repeated[0]->setElements($elements);
26.    return $repeated;
27.    }
28.    protected function get_more_choices_strin
g() {
29.    return get_string('addmoreanswerblanks',
'qtype_motorskill');
30.    }
31.    protected function add_app_id($mform) {
32.    global $DB;
33.    $id = $_GET['id'];
34.    $res = $DB->get_record("question_motorski
ll_app", array('question' => $id));
35.    $app = $res->app_id;
36.    $mform->addElement('text', 'appid', 'アプ
リのID', array('size' => 50, 'maxlength' =
> 255, 'value' => $app));
37.    $mform->setType('appid', PARAM_TEXT);
38.    $mform->addRule('appid', null, 'required',
null, 'client');
39.    }
40.    protected function add_unit_options($mform) {
41.    $mform->addElement('header', 'unithandlin
g', get_string('unithandling', 'qtype_mot
orskill'));
42.    $unitoptions = array(
43.    qtype_motorskill::UNITNONE => get_string
('onlymotorskill', 'qtype_motorskill'),
44.    qtype_motorskill::UNITOPTIONAL => get_str
ing('manymotorskill', 'qtype_motorskill
'),
45.    qtype_motorskill::UNITGRADED => get_strin
g('unitgraded', 'qtype_motorskill'),
46.    );
47.    $mform->addElement('select', 'unitrole',
get_string('unithandling', 'qtype_motorsk
ill'), $unitoptions);
48.    $penaltygrp = array();
49.    $penaltygrp[] = $mform->createElement('te
xt', 'unitpenalty', get_string('unitpenal
ty', 'qtype_motorskill'), array('size' =>
6));
50.    $mform->setType('unitpenalty', PARAM_FLOA
T);
51.    $mform->setDefault('unitpenalty', 0.10000
00);
52.    $unitgradingtypes = array(
53.    qtype_motorskill::UNITGRADEDOUTOFMARK =>
get_string('decfractionofresponsegrade',
'qtype_motorskill'),
54.    qtype_motorskill::UNITGRADEDOUTOFMAX => g
et_string('decfractionofquestiongrade', '
qtype_motorskill'),
55.    );
56.    $penaltygrp[] = $mform->createElement('se
lect', 'unitgradingtypes', '', $unitgradi
ngtypes);
57.    $mform->setDefault('unitgradingtypes',
1);
58.    $mform->addGroup($penaltygrp, 'penaltygrp
', get_string('unitpenalty', 'qtype_motor
skill'), '', false);
59.    $mform->addHelpButton('penaltygrp', 'unit
penalty', 'qtype_motorskill');
60.    $unitinputoptions = array(
61.    qtype_motorskill::UNITINPUT => get_string
('editableunittext', 'qtype_motorskill'),
62.    qtype_motorskill::UNITRADIO => get_string
('unitchoice', 'qtype_motorskill'),
63.    qtype_motorskill::UNITSELECT => get_strin
g('unitselect', 'qtype_motorskill'),
64.    );
65.    $mform->addElement('select', 'multichoice
display', get_string('studentunitanswer',
'qtype_motorskill'), $unitinputoptions);
66.    $unitsleftoptions = array(
67.    0 => get_string('rightexample', 'qtype_mo
torskill'),
68.    1 => get_string('leftexample', 'qtype_mot
orskill')
69.    );
70.    $mform->addElement('select', 'unitsleft',
get_string('unitposition', 'qtype_motors
kill'), $unitsleftoptions);
71.    $mform->setDefault('unitsleft', 0);
72.    $mform->disabledIf('penaltygrp', 'unitrol
e', 'eq', qtype_motorskill::UNITNONE);
73.    $mform->disabledIf('penaltygrp', 'unitrol
e', 'eq', qtype_motorskill::UNITOPTIONA
L);
74.    $mform->disabledIf('unitsleft', 'unitrol
e', 'eq', qtype_motorskill::UNITNONE);
75.    $mform->disabledIf('multichoicedisplay',
'unitrole', 'eq', qtype_motorskill::UNIT
NONE);
76.    $mform->disabledIf('multichoicedisplay',
'unitrole', 'eq', qtype_motorskill::UNIT
OPTIONAL);
77.    }
78.    protected function add_unit_fields($mform) {
79.    $mform->addElement('header', 'unithdr', g
et_string('units', 'qtype_motorskill'), '
');
80.    $unitfields = array($mform->createElement
('group', 'units', get_string('unitx', 'q
type_motorskill'), $this->unit_group($mfo
rm), null, false));

```



```

176. $errors['answeroptions[' . $key . ']] =
    $this->valid_answer_message($trimmedanswer);
177. }
178. if ($data['fraction'][$key] == 1) {
179.     $maxgrade = true;
180. }
181. if ($answer != '*' && !is_numeric($data
    ['tolerance'][$key])) {
182.     $errors['answeroptions[' . $key . ']] = get_
        string('xmustbemotorskill', 'qtype_motors
        kill', get_string('acceptederror', 'qtype
        _motorskill'));
183. }
184. } else if ($data['fraction'][$key] != 0 |
    |
185. !html_is_blank($data['feedback'][$key]['t
    ext'])) {
186.     $errors['answeroptions[' . $key . ']] =
        $this->valid_answer_message($trimmedanswe
        r);
187.     $answercount++;
188. }
189. }
190. if ($answercount == 0) {
191.     $errors['answeroptions[0]'] = get_string
        ('notenoughanswers', 'qtype_motorskill');
192. }
193. if ($maxgrade == false) {
194.     $errors['answeroptions[0]'] = get_string
        ('fractionsnomax', 'question');
195. }
196. return $errors;
197. }
198. protected function is_valid_answer($answe
    r, $data) {
199.     return $answer == '*' || $this->is_valid_
        number($answer);
200. }
201. protected function is_valid_number($x) {
202.     if (is_null($this->ap)) {
203.         $this->ap = new qtype_motorskill_ansi
        wer_processor(array());
204.     }
205.     list($value, $unit) = $this->ap->apply_un
        its($x);
206.     return !is_null($value) && !$unit;
207. }
208. protected function valid_answer_message
    ($answer) {
209.     return get_string('answermustbenumberorst
        ar', 'qtype_motorskill');
210. }
211. protected function validate_motorskill_op
    tions($data, $errors) {
212.     if ($data['unitrole'] != qtype_motorskil
        l::UNITNONE && trim($data['unit'][0]) ==
        '' ) {
213.         $errors['units[0]'] = get_string('unitone
            required', 'qtype_motorskill');
214.     }
215.     if (empty($data['unit']) || $data['unitro
        le'] == qtype_motorskill::UNITNONE) {
216.         return $errors;
217.     }
218.     foreach ($data['unit'] as $key => $unit)
        {
219.         if (is_motorskill($unit)) {
220.             $errors['units[' . $key . ']] =
                get_string('xmustnotbemotorskill', 'qtype
                _motorskill',
221.                 get_string('unit', 'qtype_motorskill'));
222.         }
223.         $trimmedunit = trim($unit);
224.         if (empty($trimmedunit)) {
225.             continue;
226.         }
227.         $trimmedmultiplier = trim($data['multipl
            ier'][$key]);
228.         if (empty($trimmedmultiplier)) {
229.             $errors['units[' . $key . ']] = get_stri
                ng('youmustenteramultiplierhere', 'qtype_
                motorskill');
230.         } else if (!is_motorskill($trimmedmultipl
            ier)) {
231.             $errors['units[' . $key . ']] = get_stri
                ng('xmustbemotorskill', 'qtype_motorskill
                ', get_string('multiplier', 'qtype_motors
                kill'));
232.         }
233.     }
234. }
235. $alreadyseenunits = array();
236. foreach ($data['unit'] as $key => $unit)
        {
237.     $trimmedunit = trim($unit);
238.     if ($trimmedunit == '') {
239.         continue;
240.     }
241.     if (in_array($trimmedunit, $alreadyseenun
        its)) {
242.         $errors['units[' . $key . ']] = get_stri
            ng('errorrepeatedunit', 'qtype_motorskill
            ');
243.     } else {
244.         $alreadyseenunits[] = $trimmedunit;
245.     }
246. }
247. return $errors;
248. }
249. public function qtype() {
250.     return 'motorskill';
251. }
252. }

```

2-10 Moodle プラグイン（問題の作成時や編集時にデータベースへアクセスするプログラマム）

```

1. <?php
2. defined('MOODLE_INTERNAL') || die();
3. require_once($CFG->libdir . '/questionlib
    php');
4. require_once($CFG->dirroot . '/question/t
    ype/motorskill/question.php');
5. class qtype_motorskill extends question_t
    ype {
6.     const UNITINPUT = 0;
7.     const UNITRADIO = 1;
8.     const UNITSELECT = 2;
9.     const UNITNONE = 3;
10.    const UNITGRADED = 1;
11.    const UNITOPTIONAL = 0;
12.    const UNITGRADEDOUTOFMARK = 1;
13.    const UNITGRADEDOUTOFMAX = 2;

```

```

14. public function get_question_options($question) {
15.     global $CFG, $DB, $OUTPUT;
16.     parent::get_question_options($question);
17.     if (!$question->options->answers = $DB->get_records_sql(
18.         "SELECT a.*, n.tolerance " . "FROM {question_answers} a, " . " {question_motorskill} n " . "WHERE a.question = ? " . "AND a.id = n.answer " . "ORDER BY a.id ASC", array($question->id))) {
19.         echo $OUTPUT->notification('Error: Missing question answer for motorskill question ' . $question->id . '!');
20.         return false;
21.     }
22.     $question->hints = $DB->get_records('question_hints',
23.         array('questionid' => $question->id), 'id ASC');
24.     $this->get_motorskill_units($question);
25.     $this->get_motorskill_options($question);
26.     if ($defaultunit = $this->get_default_motorskill_unit($question)) {
27.         foreach ($question->options->answers as $key => $val) {
28.             $answer = trim($val->answer);
29.             $length = strlen($defaultunit->unit);
30.             if ($length && substr($answer, -$length) == $defaultunit->unit) {
31.                 $question->options->answers[$key]->answer =
32.                 substr($answer, 0, strlen($answer)-$length);
33.             }
34.         }
35.     }
36.     return true;
37. }
38. public function get_motorskill_units($question) {
39.     global $DB;
40.     if ($units = $DB->get_records('question_motorskill_units',
41.         array('question' => $question->id), 'id ASC')) {
42.         $units = array_values($units);
43.     } else {
44.         $units = array();
45.     }
46.     foreach ($units as $key => $unit) {
47.         $units[$key]->multiplier = clean_param($unit->multiplier, PARAM_FLOAT);
48.     }
49.     $question->options->units = $units;
50.     return true;
51. }
52. public function get_default_motorskill_unit($question) {
53.     if (isset($question->options->units[0])) {
54.         foreach ($question->options->units as $unit) {
55.             if (abs($unit->multiplier - 1.0) < '1.0e-' . ini_get('precision')) {
56.                 return $unit;
57.             }
58.         }
59.     }
60.     return false;
61. }
62. public function get_motorskill_options($question) {
63.     global $DB;
64.     if (!$options = $DB->get_record('question_motorskill_options',
65.         array('question' => $question->id))) {
66.         $question->options->unitgradingtype = 0;
67.         $question->options->unitpenalty = 0.1;
68.         if ($defaultunit = $this->get_default_motorskill_unit($question)) {
69.             $question->options->showunits = self::UNITINPUT;
70.         } else {
71.             $question->options->showunits = self::UNITNONE;
72.         }
73.         $question->options->unitsleft = 0;
74.     }
75.     } else {
76.         $question->options->unitgradingtype = $options->unitgradingtype;
77.         $question->options->unitpenalty = $options->unitpenalty;
78.         $question->options->showunits = $options->showunits;
79.         $question->options->unitsleft = $options->unitsleft;
80.     }
81.     return true;
82. }
83. public function save_question_options($question) {
84.     global $DB;
85.     $context = $question->context;
86.     $oldanswers = $DB->get_records('question_answers',
87.         array('question' => $question->id), 'id ASC');
88.     $oldoptions = $DB->get_records('question_motorskill',
89.         array('question' => $question->id), 'answer ASC');
90.     $result = $this->save_units($question);
91.     if (isset($result->error)) {
92.         return $result;
93.     } else {
94.         $units = $result->units;
95.     }
96.     foreach ($question->answer as $key => $answerdata) {
97.         if (trim($answerdata) == '' && $question->fraction[$key] == 0 &&
98.             html_is_blank($question->feedback[$key]['text'])) {
99.             continue;
100.        }
101.        $answer = array_shift($oldanswers);
102.        if (!$answer) {
103.            $answer = new stdClass();
104.            $answer->question = $question->id;
105.            $answer->answer = '';
106.            $answer->feedback = '';
107.            $answer->id = $DB->insert_record('question_answers', $answer);
108.        }
109.        if (trim($answerdata) === '*') {
110.            $answer->answer = '*';
111.        } else {
112.            $answer->answer = $this->apply_unit($answerdata, $units,
113.                !empty($question->unitsleft));
114.            if ($answer->answer === false) {
115.                $result->notice = get_string('invalidmotorskillanswer', 'qtype_motorskill');
116.            }

```

```

117. }
118. $answer->fraction = $question->fraction
119. [$key];
120. $answer->feedback = $this->import_or_save
121. _files($question->feedback[$key],
122. $context, 'question', 'answerfeedback',
123. $answer->id);
124. $answer->feedbackformat = $question->feed
125. back[$key]['format'];
126. $DB->update_record('question_answers', $a
127. nswer);
128. if (!$options = array_shift($oldoptions))
129. {
130. $options = new stdClass();
131. }
132. $options->question = $question->id;
133. $options->answer = $answer->id;
134. if (trim($question->tolerance[$key]) == '
135. ') {
136. $options->tolerance = '';
137. } else {
138. $options->tolerance = $this->apply_unit
139. ($question->tolerance[$key],
140. $units, !empty($question->unitsleft));
141. if ($options->tolerance === false) {
142. $result->notice = get_string('invalidmoto
143. rskilltolerance', 'qtype_motorskill');
144. }
145. }
146. if (isset($options->id)) {
147. $DB->update_record('question_motorskill',
148. $options);
149. } else {
150. $DB->insert_record('question_motorskill',
151. $options);
152. }
153. }
154. $fs = get_file_storage();
155. foreach ($oldanswers as $oldanswer) {
156. $fs->delete_area_files($context->id, 'que
157. stion', 'answerfeedback', $oldanswer->i
158. d);
159. }
160. $DB->delete_records('question_answers', a
161. rray('id' => $oldanswer->id));
162. }
163. foreach ($oldoptions as $oldoption) {
164. $DB->delete_records('question_motorskill',
165. array('id' => $oldoption->id));
166. }
167. }
168. $result = $this->save_unit_options($quest
169. ion);
170. if (!empty($result->error) || !empty($res
171. ult->notice)) {
172. return $result;
173. }
174. $this->save_hints($question);
175. $this->save_app($question);
176. return true;
177. }
178. public function save_unit_options($questi
179. on) {
180. global $DB;
181. $result = new stdClass();
182. $update = true;
183. $options = $DB->get_record('question_moto
184. rskill_options',
185. array('question' => $question->id));
186. if (!$options) {
187. $options = new stdClass();
188. $options->question = $question->id;
189. $options->id = $DB->insert_record('questi
190. on_motorskill_options', $options);
191. }
192. }
193. if (isset($question->unitpenalty)) {
194. $options->unitpenalty = $question->unitpe
195. nalty;
196. } else {
197. $options->unitpenalty = 1;
198. }
199. $options->unitgradingtype = 0;
200. if (isset($question->unitrole)) {
201. $options->showunits = $question->unitrol
202. e;
203. if ($question->unitrole == self::UNITGRAD
204. ED) {
205. $options->unitgradingtype = $question->un
206. itgradingtypes;
207. $options->showunits = $question->multicho
208. icedisplay;
209. }
210. } else if (isset($question->showunits)) {
211. $options->showunits = $question->showunit
212. s;
213. if (isset($question->unitgradingtype)) {
214. $options->unitgradingtype = $question->un
215. itgradingtype;
216. }
217. } else {
218. if ($defaultunit = $this->get_default_mot
219. orskill_unit($question)) {
220. $options->showunits = self::UNITINPUT;
221. } else {
222. $options->showunits = self::UNITNONE;
223. }
224. }
225. $options->unitsleft = !empty($question->u
226. nitsleft);
227. $DB->update_record('question_motorskill_o
228. ptions', $options);
229. if (!empty($result->notice)) {
230. return $result;
231. }
232. return true;
233. }
234. public function save_units($question) {
235. global $DB;
236. $result = new stdClass();
237. $DB->delete_records('question_motorskill_
238. units', array('question' => $question->i
239. d));
240. if (!isset($question->multiplier)) {
241. $result->units = array();
242. return $result;
243. }
244. $units = array();
245. $unitalreadyinsert = array();
246. foreach ($question->multiplier as $i =>
247. $multiplier) {
248. if (!empty($question->multiplier[$i]) &&
249. !empty($question->unit[$i]) && !array_ke
250. y_exists($question->unit[$i], $unitalread
251. yinsert)) {
252. $unitalreadyinsert[$question->unit[$i]] =
253. 1;
254. $units[$i] = new stdClass();
255. $units[$i]->question = $question->id;
256. $units[$i]->multiplier = $this->apply_uni
257. t($question->multiplier[$i],
258. array(), false);
259. $units[$i]->unit = $question->unit[$i];
260. $DB->insert_record('question_motorskill_u
261. nits', $units[$i]);
262. }
263. }
264. unset($question->multiplier, $question->u

```

```

nit);
224. $result->units = &$units;
225. return $result;
226. }
227. public function save_app($question) {
228. global $DB;
229. $result = new stdClass();
230. $DB->delete_records('question_motorskill_
app', array('question' => $question->i
d));
231. $app = new stdClass();
232. $app->question = $question->id;
233. $app->app_id = $question->appid;
234. $DB->insert_record('question_motorskill_a
pp', $app);
235. $result->app = $app;
236. }
237. protected function initialise_question_in
stance(question_definition $question, $qu
estiondata) {
238. parent::initialise_question_instance($que
stion, $questiondata);
239. $this->initialise_motorskill_answers($que
stion, $questiondata);
240. $question->unitdisplay = $questiondata->o
ptions->showunits;
241. $question->unitgradingtype = $questiondat
a->options->unitgradingtype;
242. $question->unitpenalty = $questiondata->o
ptions->unitpenalty;
243. $question->ap = $this->make_answer_proces
sor($questiondata->options->units,
$questiondata->options->unitsleft);
244. }
245. }
246. public function initialise_motorskill_ans
wers(question_definition $question, $ques
tiondata) {
247. $question->answers = array();
248. if (empty($questiondata->options->answer
s)) {
249. return;
250. }
251. foreach ($questiondata->options->answers
as $a) {
252. $question->answers[$a->id] = new qtype_mo
torskill_answer($a->id, $a->answer,
$ a->fraction, $a->feedback, $a->feedbackf
ormat, $a->tolerance);
254. }
255. }
256. public function make_answer_processor($un
its, $unitsleft) {
257. if (empty($units)) {
258. return new qtype_motorskill_answer_proces
sor(array());
259. }
260. }
261. $cleanedunits = array();
262. foreach ($units as $unit) {
263. $cleanedunits[$unit->unit] = $unit->multi
plier;
264. }
265. return new qtype_motorskill_answer_proces
sor($cleanedunits, $unitsleft);
266. }
267. public function delete_question($question
id, $contextid) {
268. global $DB;
269. $DB->delete_records('question_motorskill'
array('question' => $questionid));
270. $DB->delete_records('question_motorskill_
options', array('question' => $questioni
d));
271. $DB->delete_records('question_motorskill_
units', array('question' => $questioni
d));
272. parent::delete_question($questionid, $con
textid);
273. }
274. }
275. public function get_random_guess_score($q
uestiondata) {
276. foreach ($questiondata->options->answers
as $aid => $answer) {
277. if ('*' == trim($answer->answer)) {
278. return max($answer->fraction - $questiond
ata->options->unitpenalty, 0);
279. }
280. }
281. return 0;
282. }
283. public function add_unit($questiondata,
$answer, $unit = null) {
284. if (is_null($unit)) {
285. $unit = $this->get_default_motorskill_uni
t($questiondata);
286. }
287. if (!$unit) {
288. return $answer;
289. }
290. if (!empty($questiondata->options->unitsl
eft)) {
291. return $unit->unit . ' ' . $answer;
292. } else {
293. return $answer . ' ' . $unit->unit;
294. }
295. }
296. public function get_possible_responses($q
uestiondata) {
297. $responses = array();
298. $unit = $this->get_default_motorskill_uni
t($questiondata);
299. $starfound = false;
300. foreach ($questiondata->options->answers
as $aid => $answer) {
301. $responseclass = $answer->answer;
302. if ($responseclass === '*') {
303. $starfound = true;
304. } else {
305. $responseclass = $this->add_unit($questio
ndata, $responseclass, $unit);
306. $ans = new qtype_motorskill_answer($answe
r->id, $answer->answer, $answer->fractio
n,
$answer->feedback, $answer->feedbackforma
t, $answer->tolerance);
307. list($min, $max) = $ans->get_tolerance_in
terval();
308. $responseclass .= " ({$min}..{$max})";
309. }
310. }
311. $responses[$aid] = new question_possible_
response($responseclass,
$answer->fraction);
312. }
313. }
314. if (!$starfound) {
315. $responses[0] = new question_possible_res
ponse(
get_string('didnotmatchanyanswer', 'quest
ion'), 0);
316. }
317. }
318. $responses[null] = question_possible_resp
onse::no_response();
319. }
320. return array($questiondata->id => $respon
ses);
321. }

```

```

322. public function apply_unit($rawresponse,
323. $units, $unitsleft) {
324. $ap = $this->make_answer_processor($units,
325. $unitsleft);
326. list($value, $unit, $multiplier) = $ap->a
327. pply_units($rawresponse);
328. if (!is_null($multiplier)) {
329. $value *= $multiplier;
330. }
331. return $value;
332. }
333. public function move_files($questionid,
334. $oldcontextid, $newcontextid) {
335. $fs = get_file_storage();
336. parent::move_files($questionid, $oldconte
337. xtid, $newcontextid);
338. $this->move_files_in_answers($questionid,
339. $oldcontextid, $newcontextid);
340. $this->move_files_in_hints($questionid,
341. $oldcontextid, $newcontextid);
342. }
343. protected function delete_files($question
344. id, $contextid) {
345. $fs = get_file_storage();
346. parent::delete_files($questionid, $contex
347. tid);
348. $this->delete_files_in_answers($questioni
349. d, $contextid);
350. $this->delete_files_in_hints($questionid,
351. $contextid);
352. }
353. }
354. class qtype_motorskill_answer_processor {
355. protected $units;
356. protected $decsep;
357. protected $thousandssep;
358. protected $unitsbefore;
359. protected $regex = null;
360. public function __construct($units, $unit
361. sbefore = false, $decsep = null,
362. $thousandssep = null) {
363. if (is_null($decsep)) {
364. $decsep = get_string('decsep', 'langconfi
365. g');
366. }
367. $this->decsep = $decsep;
368. if (is_null($thousandssep)) {
369. $thousandssep = get_string('thousandssep',
370. 'langconfig');
371. }
372. $this->thousandssep = $thousandssep;
373. $this->units = $units;
374. $this->unitsbefore = $unitsbefore;
375. }
376. public function set_characters($decsep,
377. $thousandssep) {
378. $this->decsep = $decsep;
379. $this->thousandssep = $thousandssep;
380. $this->regex = null;
381. }
382. public function get_point() {
383. return $this->decsep;
384. }
385. public function get_separator() {
386. return $this->thousandssep;
387. }
388. public function contains_thousands_seapar
389. ator($value) {
390. if (!in_array($this->thousandssep, array
391. ('.', ',', ''))) {
392. return false;
393. }
394. return strpos($value, $this->thousandsse
395. p) !== false;
396. }
397. protected function build_regex() {
398. if (!is_null($this->regex)) {
399. return $this->regex;
400. }
401. $decsep = preg_quote($this->decsep, '/');
402. $thousandssep = preg_quote($this->thousan
403. dssep, '/');
404. $beforepointre = '([+-]?[\' . $thousandsse
405. p . '\$d]*)';
406. $decimalsre = $decsep . '(\$d*)';
407. $exponentre = '(?:e|E|(?:(x|\$*|x)10(?:\$^\
408. \$*\$*)))([+-]?\$d+)'
409. $numberbit = "{$beforepointre}(?:{\$decima
410. lsre})?(?:{\$exponentre})?";
411. if ($this->unitsbefore) {
412. $this->regex = "/{\$numberbit}\$/";
413. } else {
414. $this->regex = "/^\{\$numberbit}\$/";
415. }
416. return $this->regex;
417. }
418. protected function parse_response($respon
419. se) {
420. if (!preg_match($this->build_regex(), $re
421. sponse, $matches)) {
422. return array(null, null, null, null);
423. }
424. $matches += array('', '', '', '');
425. list($matchedpart, $beforepoint, $decimal
426. s, $exponent) = $matches;
427. $beforepoint = str_replace($this->thousan
428. dssep, '', $beforepoint);
429. if ($beforepoint === '' && $decimals ===
430. '') {
431. return array(null, null, null, null);
432. }
433. if ($this->unitsbefore) {
434. $unit = substr($response, 0, -strlen($mat
435. chedpart));
436. } else {
437. $unit = substr($response, strlen($matched
438. part));
439. }
440. $unit = trim($unit);
441. return array($beforepoint, $decimals, $ex
442. ponent, $unit);
443. }
444. public function apply_units($response, $s
445. eparateunit = null) {
446. $response = str_replace(' ', '', $respons
447. e);
448. $response = preg_replace('~(?:e|E|(?:(x|\$*
449. |x)10(?:\$^\$*\$*)))([+-]?\$d+~', 'e$1', $re
450. sponse);
451. if (strpos($response, '.') !== false || s
452. ubstr_count($response, ',') > 1) {
453. $response = str_replace(',', '', $respons
454. e);
455. } else {
456. $response = str_replace('.', '.', $respon
457. se);
458. }
459. $regex = '[+-]?(?:\$d+(?:\$¥.\$d*)?|\$¥.\$d+)
460. (?:e[+-]?\$d+)?';
461. if ($this->unitsbefore) {
462. $regex = "/{\$regex}\$/";
463. } else {
464. $regex = "/^\{\$regex}\$/";
465. }
466. if (!preg_match($regex, $response, $match
467. es)) {

```

```

429. return array(null, null, null);
430. }
431. $numberstring = $matches[0];
432. if ($this->unitsbefore) {
433. $unit = (string) substr($response, 0, -str
rlen($numberstring));
434. } else {
435. $unit = (string) substr($response, strlen
($numberstring));
436. }
437. if (!is_null($separateunit)) {
438. $unit = $separateunit;
439. }
440. if ($this->is_known_unit($unit)) {
441. $multiplier = 1 / $this->units[$unit];
442. } else {
443. $multiplier = null;
444. }
445. return array($numberstring + 0, $unit, $m
ultiplier);
446. }
447. public function get_default_unit() {
448. reset($this->units);
449. return key($this->units);
450. }
451. public function add_unit($answer, $unit =
null) {
452. if (is_null($unit)) {
453. $unit = $this->get_default_unit();
454. }
455. if (!$unit) {
456. return $answer;
457. }
458. if ($this->unitsbefore) {
459. return $unit . ' ' . $answer;
460. } else {
461. return $answer . ' ' . $unit;
462. }
463. }
464. public function is_known_unit($unit) {
465. return array_key_exists($unit, $this->uni
ts);
466. }
467. public function are_units_before() {
468. return $this->unitsbefore;
469. }
470. public function get_unit_options() {
471. $options = array();
472. foreach ($this->units as $unit => $notuse
d) {
473. $options[$unit] = $unit;
474. }
475. return $options;
476. }
477. }

```

2-11 Moodle プラグイン (問題を表示するプログラム)

```

1. <?php
2. class qtype_motorskill_renderer extends q
type_renderer {
3. public function formulation_and_controls
(question_attempt $qa,
4. question_display_options $options) {
5. $question = $qa->get_question();
6. $currentanswer = $qa->get_last_qt_var('an
swer');
7. if ($question->has_separate_unit_field())
{
8. $selectedunit = $qa->get_last_qt_var('uni
t');
9. } else {
10. $selectedunit = null;
11. }
12. $inputname = $qa->get_qt_field_name('answ
er');
13. $inputattributes = array(
14. 'type' => 'text',
15. 'name' => $inputname,
16. 'value' => $currentanswer,
17. 'id' => $inputname,
18. 'size' => 80,
19. );
20. $inputattributes['readonly'] = 'disabled
';
21. $feedbackimg = '';
22. if ($options->correctness) {
23. list($value, $unit, $multiplier) = $quest
ion->ap->apply_units(
24. $currentanswer, $selectedunit);
25. $answer = $question->get_matching_answer
($value, $multiplier);
26. if ($answer) {
27. $fraction = $question->apply_unit_penalty
($answer->fraction, $answer->unitisrigh
t);
28. } else {
29. $fraction = 0;
30. }
31. $inputattributes['class'] = $this->feedba
ck_class($fraction);
32. $feedbackimg = $this->feedback_image($fra
ction);
33. }
34. global $DB;
35. $id = $qa->answers->id;
36. $id = $question->id;
37. $res = $DB->get_record("question_motorski
ll_app",array('question' => $id));
38. $app = $res->app_id;
39. $usageid = $qa->get_usage_id();
40. $questiontext = $question->format_questio
ntext($qa);
41. $questiontext .= "アプリを使って練習します。
ここで直接入力することはできません。 <br />";
42. $tempsesskey = sesskey();
43. $tempattempt = $_GET['attempt'];
44. $questiontext .= "<a href='intent://whate
ver/#Intent;scheme=". $app.";S.ssesskey=".
$tempsesskey.";S.attempt=".$tempattempt.
";S.usageid=".$usageid.";end'>アプリを起動
する</a>";
45. $placeholder = false;
46. if (preg_match('/____+/', $questiontext,
$matches)) {
47. $placeholder = $matches[0];
48. $inputattributes['size'] = round(strlen
($placeholder) * 1.1);
49. }
50. $input = html_writer::empty_tag('input',
$inputattributes) . $feedbackimg;
51. if ($question->has_separate_unit_field())
{
52. if ($question->unitdisplay == qtype_motor
skill::UNITRADIO) {
53. $choices = array();
54. $i = 1;
55. foreach ($question->ap->get_unit_options

```

```

56.     () as $unit) {
57.     $id = $qa->get_qt_field_name('unit') . '_' . $i++;
58.     $radioattrs = array('type' => 'radio', 'id' => $id, 'value' => $unit, 'name' => $qa->get_qt_field_name('unit'));
59.     if ($unit == $selectedunit) {
60.         $radioattrs['checked'] = 'checked';
61.     }
62.     $choices[] = html_writer::tag('label', html_writer::empty_tag('input', $radioattrs) . $unit, array('for' => $id, 'class' => 'unitchoice'));
63.     var_dump($id);
64.     $unitchoice = html_writer::tag('span', implode(' ', $choices), array('class' => 'unitchoices'));
65.     } else if ($question->unitdisplay == qtype_motorskill::UNITSELECT) {
66.     $unitchoice = html_writer::label(get_string('selectunit', 'qtype_motorskill'), 'menu' . $qa->get_qt_field_name('unit'), false, array('class' => 'accesshide'));
67.     $unitchoice .= html_writer::select($question->ap->get_unit_options(), $qa->get_qt_field_name('unit'), $selectedunit, array('=>'choosedots'), array('disabled' => $options->readonly));
68.     if ($question->ap->are_units_before()) {
69.         $input = $unitchoice . ' ' . $input;
70.     } else {
71.         $input = $input . ' ' . $unitchoice;
72.     }
73.     if ($placeholder) {
74.         $inputinplace = html_writer::tag('label', get_string('answer'), array('for' => $inputattributes['id'], 'class' => 'accesshide'));
75.         $inputinplace .= $input;
76.         $questiontext = substr_replace($questiontext, $inputinplace, strpos($questiontext, $placeholder), strlen($placeholder));
77.     }
78.     $result = html_writer::tag('div', $questiontext, array('class' => 'qtext'));
79.     if (!$placeholder) {
80.         $result .= html_writer::start_tag('div', array('class' => 'ablock'));
81.         $result .= html_writer::tag('label', /*get_string('answercolon', 'qtype_motorskill')*/"得点:", array('for' => $inputattributes['id']));
82.         $result .= html_writer::tag('span', $input, array('class' => 'answer'));
83.         $result .= html_writer::end_tag('div');
84.     }
85.     if ($qa->get_state() == question_state::
95.     $invalid) {
96.     $result .= html_writer::nonempty_tag('div', $question->get_validation_error(array('answer' => $currentanswer, 'unit' => $selectedunit)), array('class' => 'validationerror'));
97.     }
98.     return $result;
99. }
100. public function specific_feedback(question_attempt $qa) {
101.     $question = $qa->get_question();
102.     if ($question->has_separate_unit_field()) {
103.         $selectedunit = $qa->get_last_qt_var('unit');
104.     } else {
105.         $selectedunit = null;
106.     }
107.     list($value, $unit, $multiplier) = $question->ap->apply_units($qa->get_last_qt_var('answer'), $selectedunit);
108.     $answer = $question->get_matching_answer($value, $multiplier);
109.     if ($answer && $answer->feedback) {
110.         $feedback = $question->format_text($answer->feedback, $answer->feedbackformat, $qa, 'question', 'answerfeedback', $answer->id);
111.     } else {
112.         $feedback = '';
113.     }
114.     if ($question->unitgradingtype && !$question->ap->is_known_unit($unit)) {
115.         $feedback .= html_writer::tag('p', get_string('unitincorrect', 'qtype_motorskill'));
116.     }
117.     return $feedback;
118. }
119. public function correct_response(question_attempt $qa) {
120.     $question = $qa->get_question();
121.     $answer = $question->get_correct_answer();
122.     if (!$answer) {
123.         return '';
124.     }
125.     $response = str_replace('.', $question->ap->get_point(), $answer->answer);
126.     if ($question->unitdisplay != qtype_motorskill::UNITNONE) {
127.         $response = $question->ap->add_unit($response);
128.     }
129.     return get_string('correctansweris', 'qtype_shortanswer', $response);
130. }
131. }
132. }
133. }
134. }

```

2-12 Moodle プラグイン (学習者用アプリケーションからのデータを受け取るプログラ

ム)

```

1. <?php
2. $answers = $_GET['answers'];
3. $sesskey = $_GET['sesskey'];
4. $attempt = $_GET['attempt'];
5. $usageid = $_GET['usageid'];
6. $answer_array = explode(",", $answers);

```

```

7.  $html = "";
8.  $slots = "";
9.  for ($i=0; $i<count($answer_array); $i++)
    {
10.  $j = $i+1;
11.  $html .= "<input type='hidden' name='q".
    $usageid.".".$j."_:flagged' value='0'>";
12.  $html .= "<input type='hidden' name='q".
    $usageid.".".$j."_:sequencecheck' value='
1'>";
13.  $html .= "<input type='hidden' name='q".
    $usageid.".".$j."_answer' value='".$answe
    r_array[$i]."'>";
14.  $slots .= $j;
15.  if ($i != count($answer_array)-1) {
16.  $slots .= ",";
17.  }
18.  }
19.  $html .= "<input type='hidden' name='atte
    mpt' value='".$attempt."'>";
20.  $html .= "<input type='hidden' name='sess
    key' value='".$sesskey."'>";
21.  $html .= "<input type='hidden' name='slot
    s' value='".$slots."'>";
22.  ?>
23.  <html>
24.  <body onload='document.FRM.submit();' met
    hod='POST'>
25.  <form action='../mod/quiz/processat
    tempt.php' name='FRM'>
26.  <?php echo $html; ?>
27.  <input type='hidden' name='next' value='
    次へ'>
28.  <input type='hidden' name='thispage' valu
    e='0'>
29.  <input type='hidden' name='nextpage' valu
    e='-1'>
30.  <input type='hidden' name='timeup' value=
    '0'>
31.  <input type='hidden' name='scrollpos' val
    ue=''>
32.  </form>
33.  </body>
34.  </html>

```