筑波大学大学院博士課程

システム情報工学研究科修士論文

パーソナルクラウドを用いたテニス練習記録の蓄積、共有支援ツール

原　伊吹

修士(工学)

(コンピュータサイエンス専攻)

指導教員　田中　二郎

2013年3月

概要

近年、レンタルサーバの普及などクラウドサーバの個人使用が可能になってきたことやスマートフォンが普及してきたことにより個人でライフログを行うための技術環境が整ってきた。しかしライフログのサービス数は少なく、将来のライフログ普及には様々な課題がある。本研究ではライフログを取得、蓄積、活用の３フェーズに分け、それぞれについての課題を考える。課題を考えるツールとしてパーソナルクラウドを用いたテニス練習記録の蓄積、共有支援ツール「テニスログ」を設計、実装した。

ライフログの取得フェーズでは加速度センサとスマートフォンを用いてアンドロイドアプリケーションを設計、実装した。今後の課題としてサーバへのログ自動アップロード方式の設計などが考えられる。

ライフログの蓄積、活用フェーズでは加速度センサから取得したログをパーソナルクラウド上に蓄積させ、解析結果をグラフ表示させた。またパーソナルクラウド上にログを蓄積させた事で様々なデバイスからのアクセスが可能になり、ログの複数人共有が可能となったと考えられる。今後の課題としてログ共有から新たな知見を得るための仕組みを検討する必要があると考えられる。

目 次

第 1 章　はじめに **1**
1.1　パーソナルクラウド 1
1.2　ライフログ 1
1.2.1　ライフログの課題 2
1.2.2　ライフログのテニス練習への適用 2
1.2.3　ライフログの課題 1：記録取得フェーズ 2
1.2.4　ライフログの課題 2：記録蓄積フェーズ 3
1.2.5　ライフログの課題 3：記録活用フェーズ 3
1.3　本研究の目的 3
1.4　本論文の構成 3

第 2 章　テニスログ **4**
2.1　使用デバイス 4
2.2　ガジェットの装着 4
2.3　テニスログの使用 5
2.3.1　センサからのテニスログ取得 5
2.3.2　取得データのアップロード 10
2.3.3　センサデータの解析結果表示 11
2.3.4　アップロードデータをグラフで見る 12
2.4　利用シナリオ 14

第 3 章　テニスログのシステム概要 **25**
3.1　システム全体概要 25
3.2　センサデータの取得 25
3.3　センサデータのアップロード 32
3.4　センサデータの解析 33
3.4.1　打球数検出アルゴリズム 34
3.4.2　練習時間検出アルゴリズム 35
3.5　パーソナルクラウドへのデータ保存 36
3.5.1　生データの保存 36
3.5.2　解析データの保存 42
3.6　センサデータの活用-グラフ表示- 43

第 4 章　関連研究 **46**
4.1　ライフログを共有する Life Networking Service 46
4.2　Smart Makeup System 46
4.3　Life Log Media 47
4.4　レシートログ 47
4.5　テニススウィングの解析と定量評価 47
4.6　SlamTracker 47
4.7　画像処理を用いたテニス競技のデータ自動記録・分析手法 48
4.8　テニス試合におけるポイントごとの動画シーン抽出システム 48

第 5 章　まとめと今後の課題 **49**
5.1　ライフログ取得フェーズのまとめと今後の課題 49
5.2　ライフログ蓄積、活用フェーズのまとめと今後の課題 49
5.3　考察 49

謝辞 **51**

参考文献 **52**

図目次

2.1 加速度センサ「WAA010」 5
2.2 スマートフォン「Nexus S」 5
2.3 ガジェット装着前 6
2.4 ガジェット装着後 6
2.5 加速度センサをリストバンドで固定 7
2.6 テニスロガー起動画面 7
2.7 ペアリングデバイス候補 7
2.8 ペアリングデバイスの選択 8
2.9 ペアリングデバイス接続中 8
2.10 ペアリング完了 8
2.11 センサ計測開始コマンドの送信 8
2.12 センサデータを受信中 9
2.13 センサ計測停止コマンドの送信 9
2.14 センサ計測停止 9
2.15 「スタート！」ボタンでセンサ計測再開 9
2.16 「テニスログホーム」の起動画面 11
2.17 アストロファイルマネージャ 11
2.18 端末内に保存されたセンサデータ 12
2.19 センサデータのアップロード完了画面 12
2.20 グラフ表示への自動ジャンプ画面 14
2.21 スマートフォンでのグラフ表示 15
2.22 グラフのシングルユーザ表示モード (左半分) 16
2.23 グラフのシングルユーザ表示モード (右半分) 17
2.24 円グラフにカーソルを合わせる 18
2.25 棒グラフにカーソルを合わせる 19
2.26 折れ線グラフにカーソルを合わせる 20
2.27 ストローク打球数データのみの表示 21
2.28 練習時間データの表示 21
2.29 グラフのズーム機能：１ヶ月分のみ表示 22
2.30 グラフのマルチユーザ表示モード 22
2.31 マルチユーザ表示モードで 1ヶ月分のデータを表示 23

2.32 10 月より Bob のボレー練習量が増加 23

3.1 テニスログのシステム全体概要 26
3.2 テニスログホームのプログラム構成 27
3.3 センサデータファイル (CSV 形式) の中身 29
3.4 加速度センサの装着方向 30
3.5 ユーザ ID 入力エラー 33
3.6 ファイルアップロードエラー 33
3.7 ストローク：テークバック 36
3.8 ストローク：インパクト前 36
3.9 ストローク：インパクト 37
3.10 ストローク：フォロースルー 37
3.11 ストローク：XYZ 軸加速度データ 37
3.12 ストローク：X 軸加速度データ 38
3.13 ストローク：Y 軸加速度データ 38
3.14 ストローク：Z 軸加速度データ 39
3.15 ボレー：テークバック 39
3.16 ボレー：インパクト 39
3.17 ボレー：フォロースルー 39
3.18 ボレー：X 軸加速度データ 40
3.19 サーブ：テークバック 40
3.20 サーブ：インパクト前 40
3.21 サーブ：インパクト直前 41
3.22 サーブ：フォロースルー 41
3.23 サーブ：X 軸加速度データ 41
3.24 ユーザごとのパーソナルフォルダ 42
3.25 パーソナルフォルダ内のセンサ生データファイル 42
3.26 解析データが保存されてる MySQLDB テーブル 43
3.27 シングルユーザ表示での ID 入力エラー 45
3.28 マルチユーザ表示での ID 入力エラー 45

第1章　はじめに

1.1　パーソナルクラウド

パーソナルクラウドとはクラウドコンピューティングサービスを個人向けに提供するサービスである [1]。クラウドコンピューティングとはインターネットを介し、サーバ上のストレージリソースやアプリケーションを提供するサービスである [2]。これまではクラウドサービスは主に法人向けで、個人が利用する事は難しかった。しかし近年、安価なレンタルサーバの普及など個人でクラウドを使用できる環境が整ってきた。個人のライフログデータをクラウドサーバに保存するメリットとしてデスクトップ PC、携帯情報端末など様々なデバイスからログにアクセスできるという事が挙げられる [3][4]。近年、個人向けにクラウドコンピューティングサービスを提供する様々なパーソナルクラウドサービスが展開されている。Facebook[5] や Evernote[6] などは個人向けにパーソナルデータをクラウドで扱うサービスを提供しているため、パーソナルクラウドサービスの一例として捉えることができる。

1.2　ライフログ

ライフログとは生活のある場面をデジタルで記録し続ける概念の事である。近年、スマートフォンなどの携帯情報端末や安価で小型なセンサ機器など、ライフログを取得できる様々なデバイスが登場してきた [7][8]。またライフログを保存するためのストレージとしてパーソナルクラウドサービスを利用する事が可能となってきた [9]。これら情報技術の発展により、センサを使った行動ログ取得 [10] など、個人でライフログを取得、蓄積、活用する事が可能となってきている [11][12]。

ライフログの研究としてマイライフビッツ [13]、センサカム [14]、食事ログ [15] などが挙げられる。マイライフビッツでは日常生活において発生する電話、電子メール、請求書などあらゆるデータをまるごとハードディスクに記憶させる。センサカムでは首から下げたカメラが常に写真を撮り、ユーザの行動を記録させる事が出来る。食事ログでは毎回の食事を写真に撮り、保存する事で食生活を記録している。このように食事記録や行動記録などライフログを実際に行える技術環境が整ってきているため、今後は生活の様々な場面においてライフログが行われると考えられる。

1.2.1　ライフログの課題

技術的には個人でのライフログは可能だが、現在普及しているライフログサービス数は少ない。ライフログサービスの今後の課題を考える方法として以下の 2 通りが考えられる。

1 : 既存のライフログシステムを調査して考える

2 : 新規にライフログシステムを構築しながら考える

徳永ら [16] はレシートログシステムを構築する事によりライフログサービスについての考察を行っている。本研究では第 4 章で挙げる関連研究のライフログシステムなどを参考に、徳永らと同様、主に 2 の手法を採用することにした。

1.2.2　ライフログのテニス練習への適用

生活場面には衣食住やスポーツ、趣味など様々なカテゴリー、種類がある。本研究ではライフログの対象としてテニス練習を選んだ。近年、スポーツと画像解析を融合させた研究 [17] やユビキタス環境をスポーツに適用させるなど、スポーツとコンピュータ技術を組み合わせた様々な研究が行われている [18][19][20]。また実際にユーザが使えるサービスとしてスマートフォンとセンサーを利用した NIKE+[21] などが登場してきている。

ライフログの題材について具体的にテニス練習記録の取得、蓄積、活用法を考えることにより、ライフログの課題を考える。

テニス練習から取得するデータ

テニス練習には主に技術トレーニング、フィジカルトレーニング、メンタルトレーニングにがある。技術トレーニングではラリー、球出し、ゲームなどを行う。フィジカルトレーニングでは筋肉トレーニングやランニング、ストレッチなどを行う。メンタルトレーニングでは試合本番の緊張想定などを行う。

本システムでは技術トレーニングにおける「ストローク」「ボレー」「サーブ」「練習時間」を扱う。　さらにテニス練習量を定量的に記録するデータとして「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」を用いる。定量的に記録する事で日々の練習の積み重ねを他人と同じものさしで比較する事ができる。本論文におけるテニス練習量とはこの「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」を指す。

1.2.3　ライフログの課題 1：記録取得フェーズ

ライフログの記録取得フェーズの課題としてタッチパネルなどのログ手動入力はユーザに負荷がかかるという課題が挙げられる [22]。我々は先行研究 [22] においてライフログの継続にはモチベーションを保つ事が重要だと考え、ノートへのコメント記録といった定性的記録

手法からスマートフォンを使った定量的記録手法を提案し、実装した。しかしスマートフォンによるログのタッチパネル入力においてもユーザは意識してログ入力を行うため負荷がかかるいえる。

1.2.4　ライフログの課題 2：記録蓄積フェーズ

ライフログの記録蓄積フェーズの課題として様々なデバイスからアクセスできるようログをクラウドサーバに保存するという事が挙げられる。[23]。Kim ら [23] は GPS やマイクロホン、ボディセンサからユーザの日常生活にけるログを携帯情報端末を中継して取得し、ログを Windows XP 上にに作成したサーバに蓄積させている。

1.2.5　ライフログの課題 3：記録活用フェーズ

ライフログの記録活用フェーズの課題としてクラウドサーバへ蓄積したログよりユーザの過去を想起させ、知見を得るという事が挙げられる [24]。Koh ら [24] の「KiokuHacker」では写真とジオタグを Twitter のタイムライン形式で表示させ、過去を想起させている。

1.3　本研究の目的

本研究の目的はパーソナルクラウドを用いたテニス練習記録の蓄積、共有支援ツール「テニスログ」を使ってライフログの課題を考える事である。ライフログの取得、蓄積、活用の３フェーズそれぞれについての課題を考える。本研究ではログの取得に加速度センサ、スマートフォンを用いて取得フェーズの課題を考える。またログの蓄積、活用にパーソナルクラウドの概念、技術を用いて蓄積、活用フェーズの課題を考える。

1.4　本論文の構成

本論文の構成を以下に述べる。第２章では本研究で使用するために設計、実装したツール「テニスログ」を紹介する。第３章では「テニスログ」のシステム概要について述べる。第４章では関連研究について述べる。第５章にてまとめと考察を述べる。

第2章　テニスログ

2.1　使用デバイス

テニスログでは練習量データを取得するデバイスと練習量データを蓄積、共有するためのパーソナルクラウドサーバを使用する。

練習量データを取得するデバイスには ATR-Promotions 社の小型無線ハイブリッドセンサ「WAA010」[25] とアンドロイド OS を搭載したスマートフォン「Nexus S」[26] を使用する。

小型無線ハイブリッドセンサには３軸加速度センサ、３軸角速度センサ、地磁気センサが内蔵されている。本研究ではテニスログを取得するために３軸加速度センサのみを使用する。テニスログではスイングの加速度変化データを利用して打球数記録を行う。これ以後「WAA010」を加速度センサと呼ぶ。またこれ以後「Nexus S」をスマートフォンと呼ぶ。図 2.1 に「WAA010」を、図 2.2 に「Nexus S」の画像を示す。

パーソナルクラウドサーバには BIGLOBE クラウドホスティングサービス [27] を利用する。パーソナルクラウドにはセンサより取得した生データと生データ解析により抽出した「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」データを蓄積させる。

2.2　ガジェットの装着

図 2.3 と図 2.4 にテニスログ取得デバイスの装着前、装着後画像を示す。図 2.3 では右手に加速度センサを、左手にスマートフォンを持っている。図 2.4 では右手首のリストバンド内に加速度センサを、左スボンポケットにスマートフォンを装着している。図 2.5 において図 2.4 の手首部分を拡大した画像を示す。リストバンドと手首の間に加速度センサが入っているため、図 2.5 においてリストバンドの加速度センサ部分が膨らんでいる。

スマートフォンをズボンのポケットに入れる理由はセンサとスマートフォンを通信圏内に常においておくためである。加速度センサは常にスマートフォンに計測データを送っており、無線通信範囲は 10m 以内である。日本テニス協会のルールブック [28] によるとテニスコートの広さはダブルスのラインも含めると縦 23.77m、幅 10.97m である。そのためテニスコート外のベンチなどにスマートフォンを置くと、無線通信範囲からセンサが外れてしまう可能性がある。

図 2.1: 加速度センサ「WAA010」

図 2.2: スマートフォン「Nexus S」

2.3 テニスログの使用

2.3.1 センサからのテニスログ取得

加速度センサの無線通信規格は Bluetooth である。センサ計測の開始や停止など加速度センサの操作は外部よりテキストコマンドを受信する事で行う。本研究ではセンサ操作、センサデータ受信、受信データの端末内保存操作を行うためのアンドロイドアプリケーション「テニスロガー」を設計、実装した。テニスロガーをスマートフォンにインストールし、使用する事で加速度センサからテニスログの取得を行う。

図 2.6〜図 2.15 はテニスロガーの使用画面である。図 2.6〜図 2.10 は加速度センサとスマートフォン間でのペリング処理の様子を、図 2.11〜図 2.15 はセンサ計測開始コマンドの送信から停止までの様子を示している。

Bluetooth ペアリング処理

図 2.6 はテニスロガーの起動画面である。アプリを立ち上げると初めにこの画面が表示される。加速度センサとの Bluetooth ペアリング処理を行うため、ユーザは画面左下の「端末検索」ボタンを押す。図 2.7 は「端末検索」ボタンを押した後に表示される画面である。Bluetooth でペアリングするデバイスの候補が示される。図 2.8 に加速度センサを選択する様子を示す。

図 2.3: ガジェット装着前

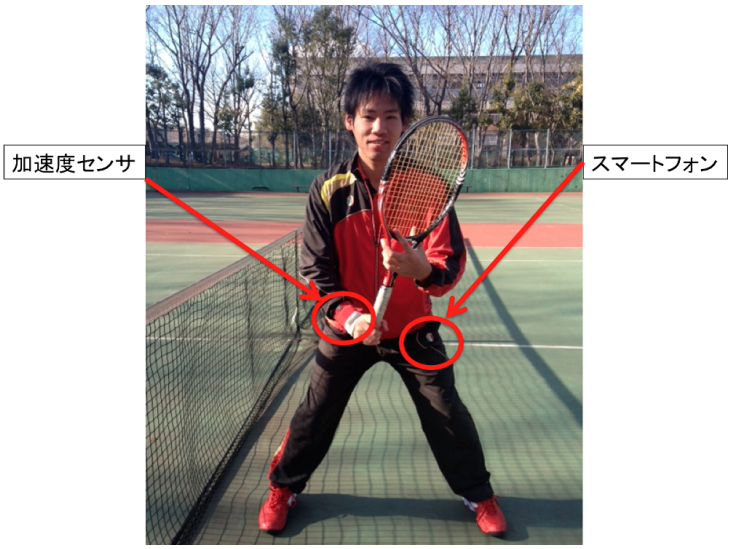

図 2.4: ガジェット装着後

図 2.8 では加速度センサである「WAA010-0505」を選択している。図 2.9 にペアリングデバイス接続中の画面を示す。ペアリングデバイスを選択すると「テニスロガー」は加速度センサに接続要求を行い、ペアリングが完了するまでユーザは待機する。図 2.10 にペアリング完了を通知する画面を示す。ペアリングが完了すると「センサ接続完了！「スタート！」ボタンを押すとテニスログが開始されます！」と表示され、ユーザは「スタート！」ボタンを押す事でセンサ計測を開始する事が出来る。

センサ計測

図 2.11 に加速度センサに計測開始命令を行う様子を示す。「スタート！」ボタンを押すと計測開始コマンドがセンサに送信され、ユーザは加速度データの受信を待つ。図 2.12 に加速度データを受信している様子を示す。「Now Logging ･･･ 」と表示され、加速度センサよりデータを受信している。計測中は常にスマートフォンに加速度データを転送し続ける。スマートフォンではテニスロガーを経由して端末内に加速度データが保存されていく。テニスロガーは「アプリ End」ボタンを押すまで常に起動し続けている。センサデータを受信している間もバックグラウンドで動き続ける事ができる。そのためテニスロガーでセンサデータを受信しながら他のアプリを操作する事も可能である。

図 2.13 にセンサ計測停止命令を行う様子を示す。「ストップ！」ボタンを押すとセンサ計測

図 2.5: 加速度センサをリストバンドで固定

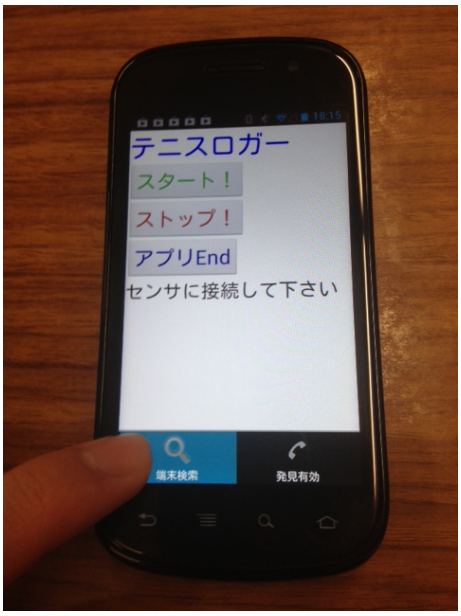

図 2.6: テニスロガー起動画面

図 2.7: ペアリングデバイス候補

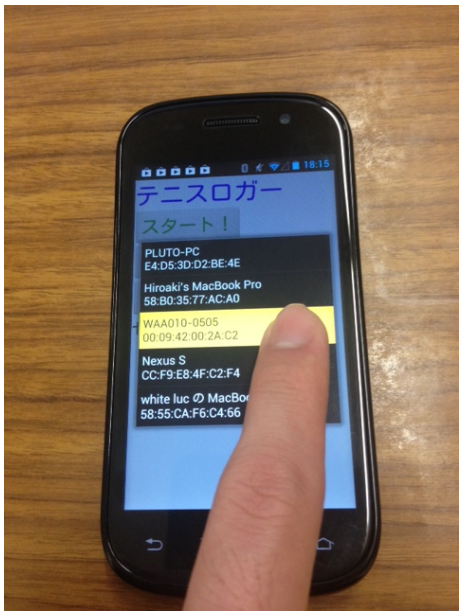

図 2.8: ペアリングデバイスの選択

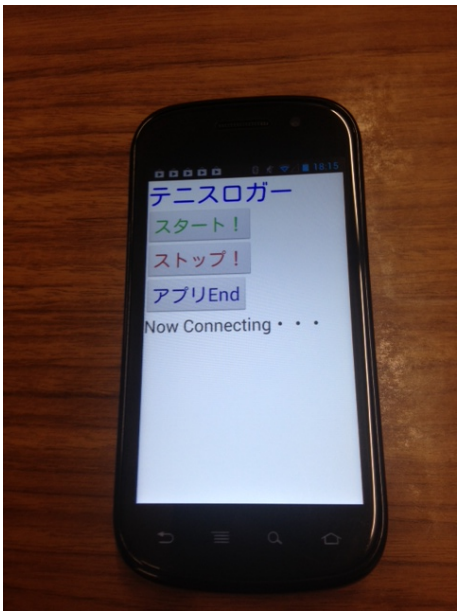

図 2.9: ペアリングデバイス接続中

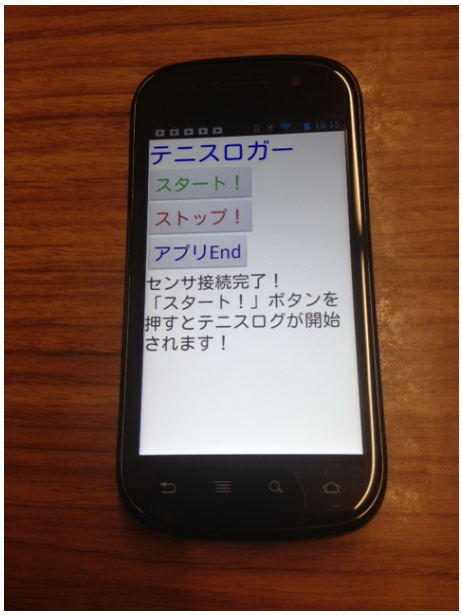

図 2.10: ペアリング完了

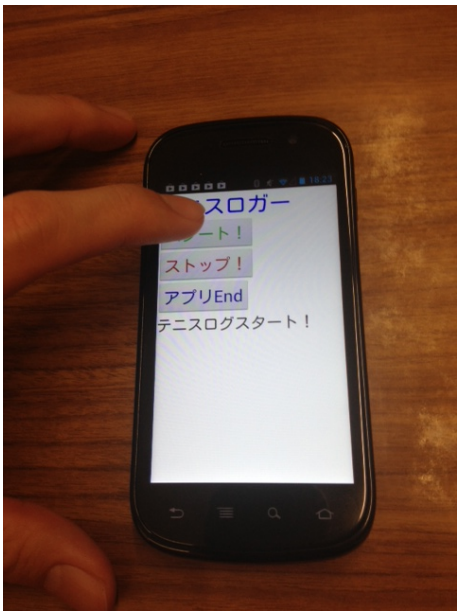

図 2.11: センサ計測開始コマンドの送信

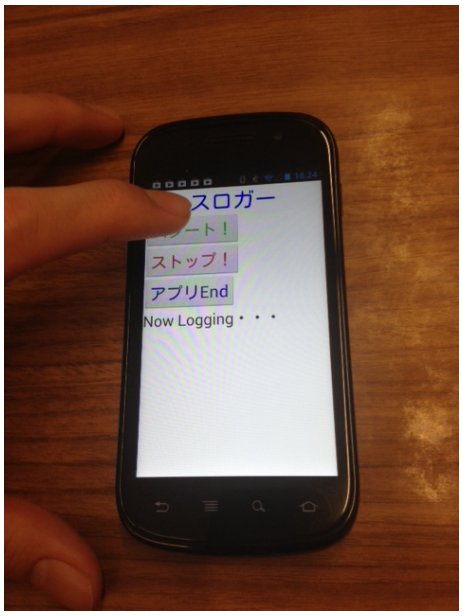

図 2.12: センサデータを受信中

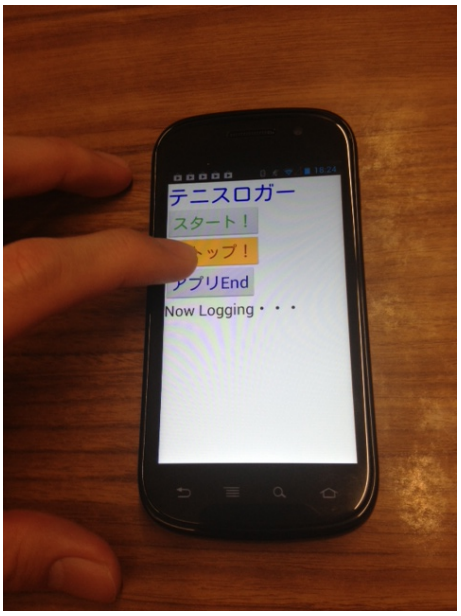

図 2.13: センサ計測停止コマンドの送信

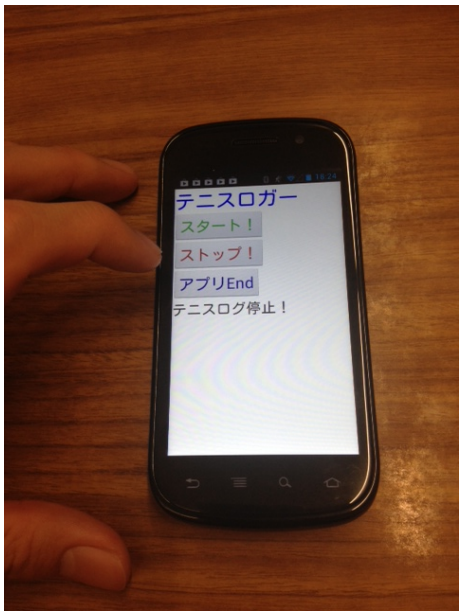

図 2.14: センサ計測停止

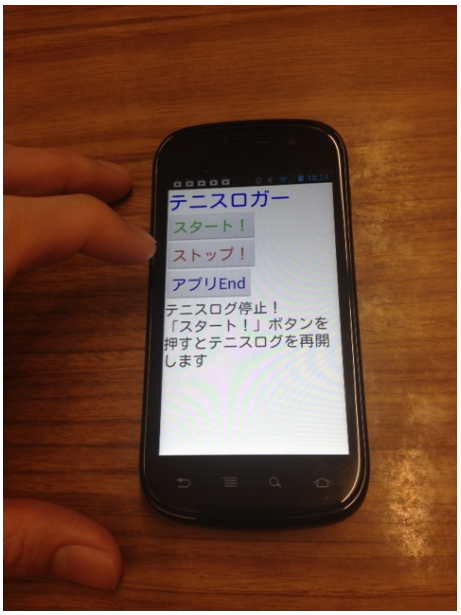

図 2.15:「スタート！」ボタンでセンサ計測再開

停止コマンドがセンサに送信され、センサ計測を停止する。図 2.14 にセンサ計測が停止した様子を示す。「テニスログ停止！」と表示されるとセンサ計測停止コマンドが送られた事を示す。図 2.15 にセンサ計測停止が無事完了した画面を示す。センサ計測停止コマンドがセンサに受け付けられ、無事計測停止すると「テニスログ停止！「スタート！」ボタンを押すとテニスログを再開します」と表示される。「スタート！」ボタンを押す事でセンサ計測を再開する事ができる。また「アプリ End」ボタンを押すと加速度センサとのペアリングも自動的に解除され、アプリは終了する。

「アプリ End」ボタンが押されるまでセンサデータは同一ファイル内に保存される。つまり、一度「ストップ！」ボタンを押し、「アプリ End」ボタンを押さず「スタート！」ボタンを押した場合、計測が停止しても同じファイル内にデータは保存される。練習以外で計測を停止させる事により、無駄なデータの取得を節約する事が出来る。例えばこの機能はテニス練習中の休憩時に使用できる。休憩開始時に「ストップ！」ボタンを押し、練習再開後「スタート！」ボタンを押す事で無駄なデータ取得を行わずに済む。

2.3.2 取得データのアップロード

図 2.16〜図 2.19 に取得データのアップロード操作の様子を示す。我々は取得データのアップロード、取得データの解析結果表示ツールとして Web アプリ「テニスログホーム」を設計、実装した。スマートフォンよりインターネットに接続するため「ブラウザ」を起動し、「テニスログホーム」の URL を打ち込み、Web アプリを起動する。図 2.16 に「テニスログホーム」の起動画面を示す。テニスログは Web アプロであり URL を入力するとこの画面がはじめに表示される。取得データのアップロード操作には画面一番上の項目「テニスログをアップロード」を使用する。まず任意のユーザ ID を半角英数字で入力する。図 2.16 では「ibuki」と入れている。次に ID 入力欄の下にある「ファイルを選択」ボタンを押す。図 2.17 にアップロードに使用するアプリ選択画面を示す。ID を入力すると図 2.17 のようにファイルアップロードに使用するアプリケーション選択画面が開く。テニスロガーでは取得データを端末内の SD カードに保存しているため、SD カード内のデータを参照できる「アストロファイルマネージャ」を使用する。「アストロファイルマネージャ」は画像、音楽、ビデオやドキュメントを管理するアプリケーションである。本研究で使用するため GooglePlay[29] よりダウンロードし, スマートフォンにインストールした。

図 2.18 にセンサデータファイルの一覧を表示する画面を示す。図 2.18 はアストロファイルマネージャにより SD カード内のセンサデータが保存されているフォルダ内のファイル一覧である。白いファイルアイコン一つが一つのセンサデータファイルである。この中からアップロードするセンサデータを選択する。データは CSV 形式のファイルで保存されており、ファイル名にセンサデータの取得日時を使用している。ユーザはファイル名の日時を利用してアップロードしたいセンサデータを選択する。選択後、図 2.16 において「パーソナルクラウドへ送信」ボタンを押す。図 2.19 にセンサデータのアップロード完了画面を示す。「パーソナルクラウドへ送信」を押すと図 2.19 においてセンサデータのアップロード処理が無事完了した旨を伝えるメッセージが表示される。図 2.19 では下記のメッセージを表示している。

ibuki さんのテニスログアップロード Complete!
アップロード内容
ファイル名：ibuki-2012_12_21-20_19_17.csv
ストローク打球数：95 球
ボレー打球数：93 球
サーブ打球数：球
練習時間：99.8166(省略) 分
データ破損率：0%

スマートフォンからサーバへログのアップロードに要した時間を計測した。4.8MB のデータで 4.5 秒かかった。

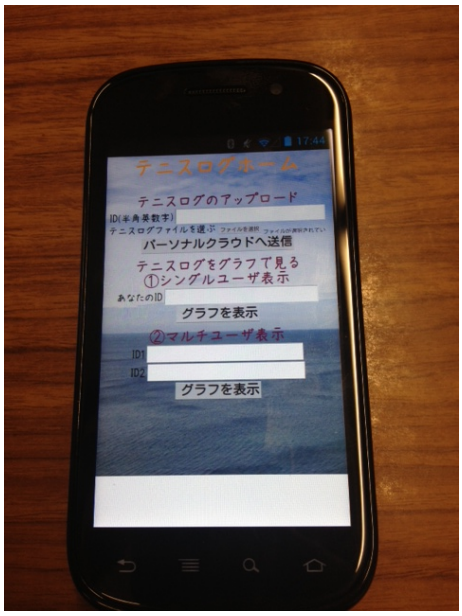

図 2.16: 「テニスログホーム」の起動画面

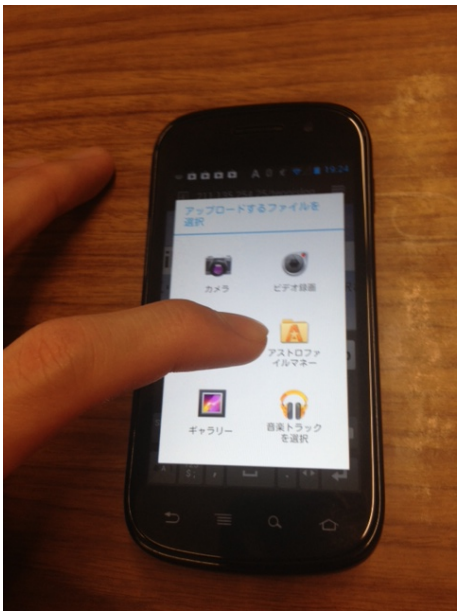

図 2.17: アストロファイルマネージャ

2.3.3 センサデータの解析結果表示

図 2.19 のアップロード完了画面ではセンサデータの解析結果を同時に表示している。表示内容は「アップロードファイル名」「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」「データ破損率」である。

アップロードファイル名とは日時を使用したセンサデータのファイル名の先頭にユーザ名 (ユーザ ID) を加えたものである。日時だけを使用してファイル名をパーソナルクラウドへ保存した場合、他ユーザと被る可能性がある。ユーザ ID をファイル名に加える事でパーソナルクラウド上ではユニークなファイル名として保存する事ができる。

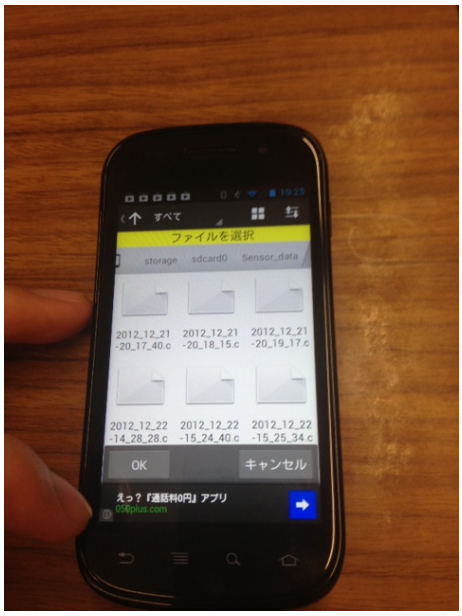

図 2.18: 端末内に保存されたセンサデータ

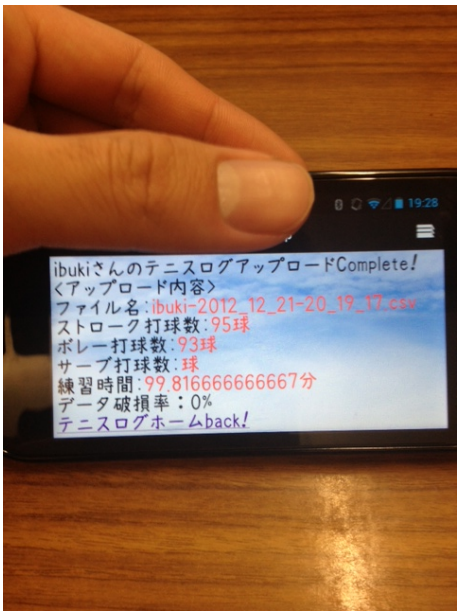

図 2.19: センサデータのアップロード完了画面

テニスログホームではスマートフォンからのセンサデータアップロード処理過程において同時にセンサデータの解析を行っている。解析処理により「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」を算出し、結果を表示している。　データ破損率とは加速度センサ、スマートフォン間での通信処理におけるセンサデータの破損具合を確認する指標である。図 2.19 においては破損率が 0%であることから加速度センサ、スマートフォン間の通信は無事正常に行われた事を示す。

2.3.4　アップロードデータをグラフで見る

テニスログではアップロードしたセンサデータの解析結果をグラフで一覧表示する事が出来る。図 2.16 においてアップロード時に使用した ID を入力し、「グラフを表示」ボタンを押すと図 2.20 の画面にジャンプする。図 2.20 では「2 秒後にグラフが自動表示されます …」というメッセージを表示させ、2 秒後にグラフが表示される事をユーザに通知している。グラフ表示には２種類のモードがあり、「シングルユーザ表示」と「マルチユーザ表示」がある。「シングルユーザ表示」は自分の過去の練習記録のみを振り返りたい場合に使う。「マルチユーザ表示」はダブルスパートナーの練習量など友人の練習量を見たり、自分と友人の練習量を比較したい場合などに使う。

シングルユーザ表示

図 2.21、図 2.22、図 2.23 に ID「Bob」でグラフをシングルユーザ表示した様子を示す。図 2.21 はスマートフォンでグラフを参照している様子である。スマートフォンの Web ブラウザからテニスログの URL へアクセスし、表示させている。図 2.22、図 2.23 は見やすいよう PC ブラウザで同じ画面を参照、拡大した画像である。これ以降のグラフ画像は PC ブラウザで参照、拡大した画像を使用する。図 2.21 では横軸に日付を取り、9/01〜10/30 までの練習量データを示している。グラフでは練習量データである「ストローク打球数」「ボレー打球数」「サーブ打球数」「練習時間」を 1 つの画面内で示している。「ストローク打球数」は青色棒グラフ、「ボレー打球数」は緑色棒グラフ、「サーブ打球数」はピンク色棒グラフ、「練習時間」は赤色折れ線グラフで示している。打球数データについては左縦軸 (単位：球)、練習時間データについては右縦軸 (単位：分) にメモリをとっている。

左上の円グラフは打球数データについて球種別に過去の合計打数記録を示している。円グラフの色と棒グラフの色は対応しており、同じ球種である。

円グラフ、棒グラフ、折れ線グラフはマウスカーソルを合わせると具体的数値が表示される。図 2.24、図 2.25、図 2.26 にデータの具体的数値を表示させている様子を示す。図 2.24 では円グラフのストローク部分にカーソルを合わせ、過去のストロークの合計打球数が 5800 球である事を示している。図 2.25 では棒グラフのストローク打球数部分にカーソルを合わせ、9/22 のストローク打球数が 510 球である事を示している。図 2.26 では 9/27 の折れ線グラフにカーソルを合わせ、練習時間が 170 分である事を示している。

また棒グラフでは 1 種類のデータのみを見たい場合などに利用できる球種別表示機能が備わっている。横軸メモリの日付下にある色ラベルの意味を示している箇所において、表示を消したいラベルをクリックする事で単一データの表示ができる。図 2.27、図 2.28 に「ストローク打球数」「練習時間」データのみをそれぞれ単一表示させた様子を示す。図 2.27 ではボレー打球数、サーブ打球数、練習時間データを消し、ストローク打球数データである青棒グラフのみを示している。図 2.28 では打球数データを消し、練習時間データである折れ線グラフのみを示している。この機能は「ボレー打球数」「サーブ打球数」においても表示可能である。なお縦軸のメモリ幅は自動的に適切な幅に切り替わる。

その他の機能としてグラフの一部をマウスカーソルで選択することにより選択範囲のグラフをズームする機能も備わっている。図 2.29 に選択範囲をズーム操作する様子を示す。図 2.29 では 9/01〜9/29 までの 1 ヶ月部分をズーム表示している。この時、ズームした縦軸の目盛は自動的に適切な幅に変わる。

マルチユーザ表示

図 2.30 にマルチユーザ表示モードのグラフを示す。図 2.30 ではユーザ ID「Bob」「John」の 9/01〜10/30 までの練習量データを縦に並べて表示させている。図 2.16 の「マルチユーザ表示」項目において比較したいユーザ ID を 2 つ入力し、「グラフを表示」ボタンを押すと見る事ができる。マルチユーザ表示においてもシングルユーザ表示において使用する機能と同じ

機能がそれぞれのグラフで使える。図 2.31 にマルチユーザ表示モードにおいてズーム機能を使用した様子を示す。図 2.31 では「Bob」と「John」の 9/01〜9/29 の練習量データを示している。

このようにマルチユーザ表示モードではグラフのズーム機能、球種別表示機能を利用する事で過去の練習量データの中から特定範囲データの比較を行う事が出来る。

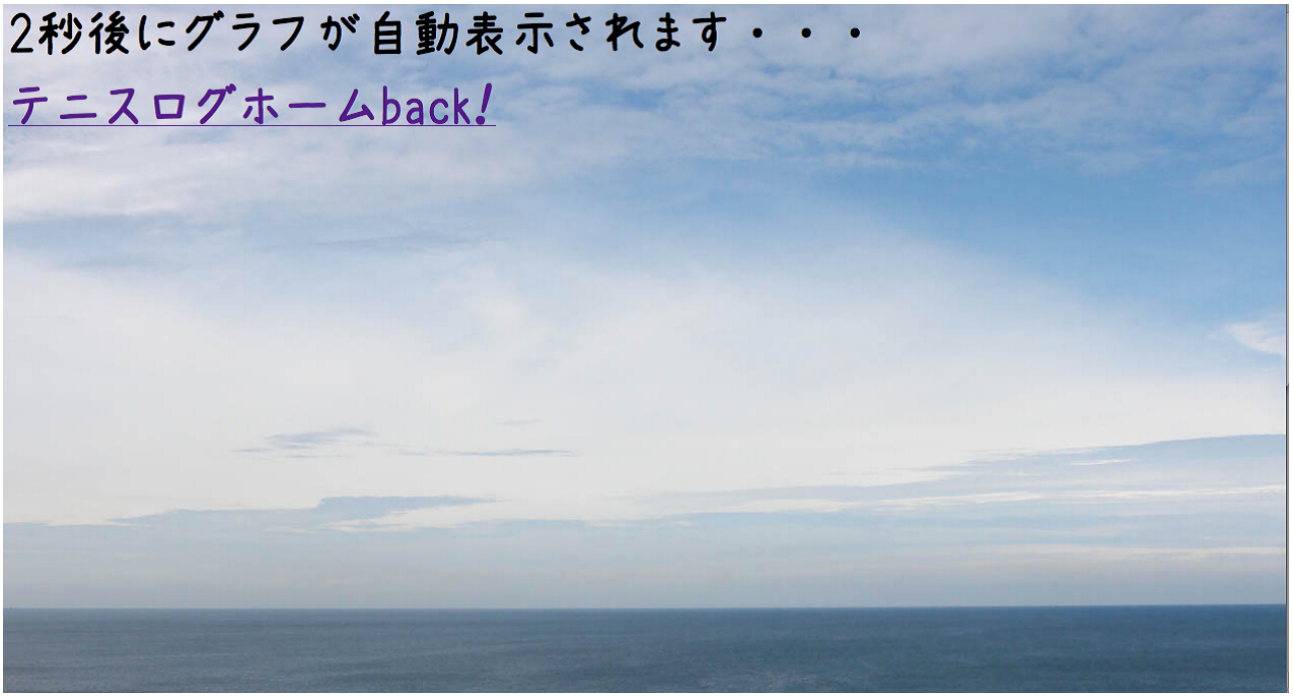

図 2.20: グラフ表示への自動ジャンプ画面

2.4 利用シナリオ

Bob と John は大学 3 年生であり、大学入学時から同じテニスサークルに所属している。２人は仲がよく 1 年生からダブルスのペアを組み、現在も部内戦やテニストーナメントなど様々な試合に出てテニスを楽しんでいる。そのため練習量も多く、数時間の練習を週に 3 回程行っている。また週 3 回の練習のうち 1 回は２人でダブルスを合わせようと２人で練習する事にしている。John は 8 月の末日、いつものように Bob に 11 月 1 日の草トーナメントに出ようと話をもちかけ、２人でダブルス大会に出る事を決めた。２人が出場を決めたちょうどその頃クラウドサービスを展開する大手 IT ベンダー AIBM 社とテニスラケットで有名な YOONEX 社が共同で新製品「テニスログ」を一般ユーザ向けに加速度センサとセットで販売し始めた。

Bob と John の２人はお互いのテニス練習についてよく話をしていた。しかし２人ともお互いが離れてるときにどれくらい練習しているかは分からず、定量的、具体的に知る方法はな

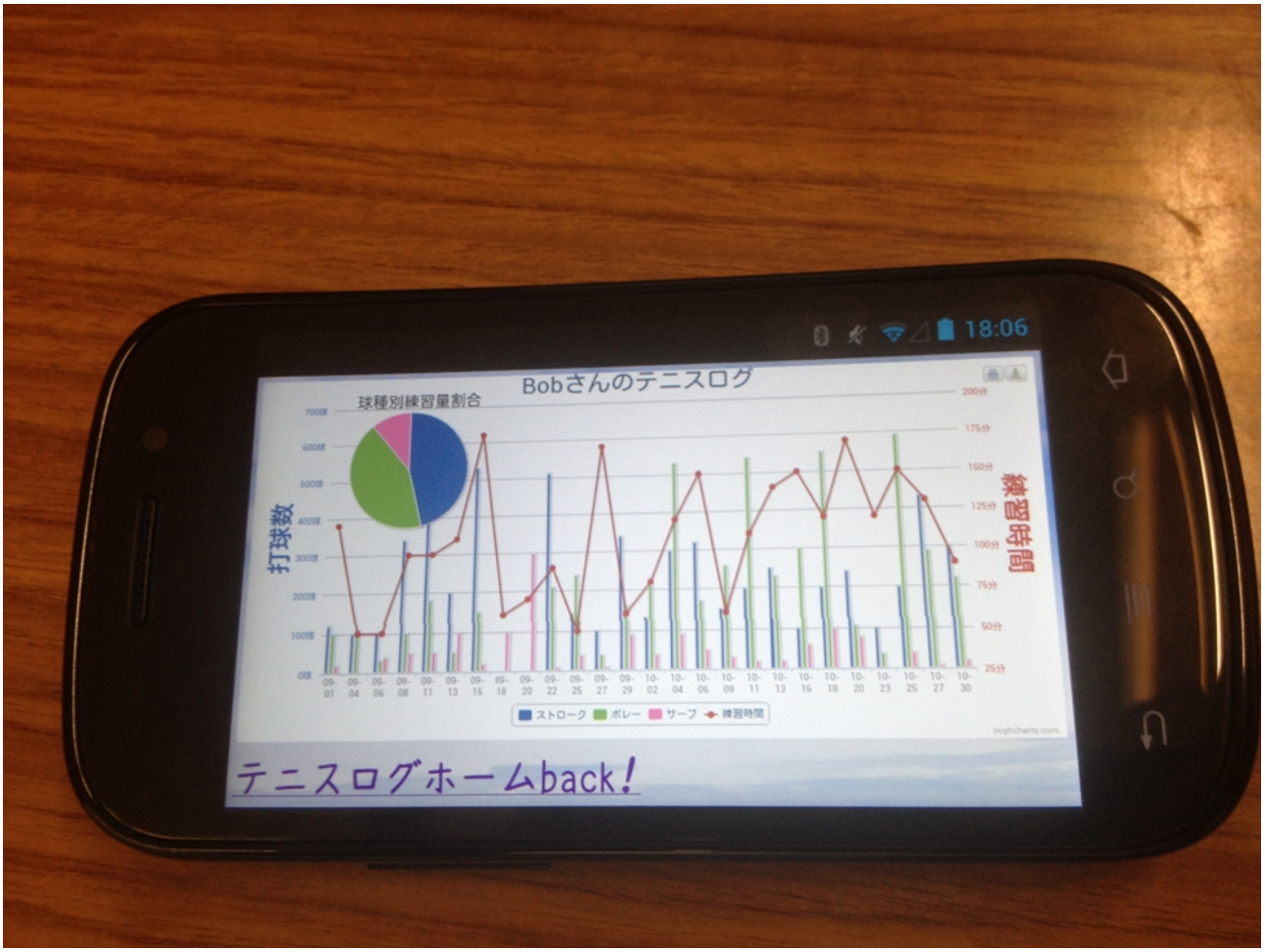

図 2.21: スマートフォンでのグラフ表示

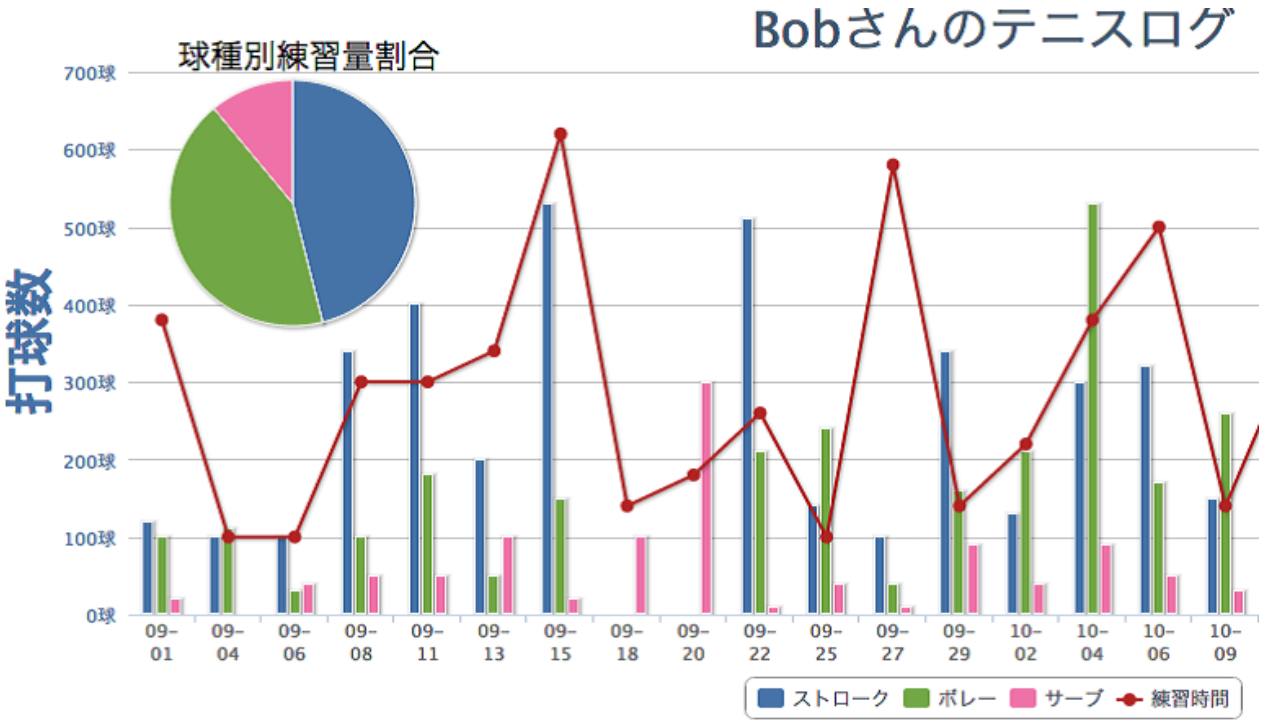

図 2.22: グラフのシングルユーザ表示モード (左半分)

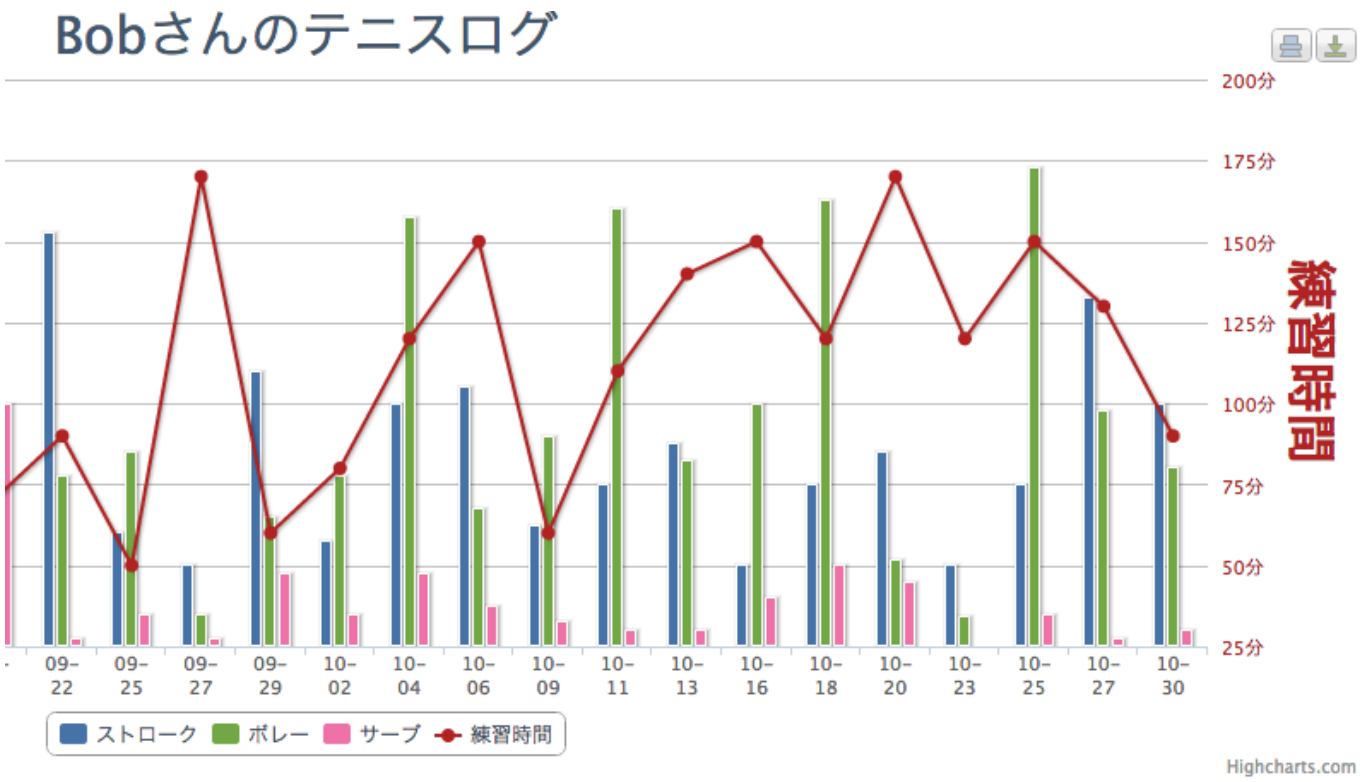

図 2.23: グラフのシングルユーザ表示モード (右半分)

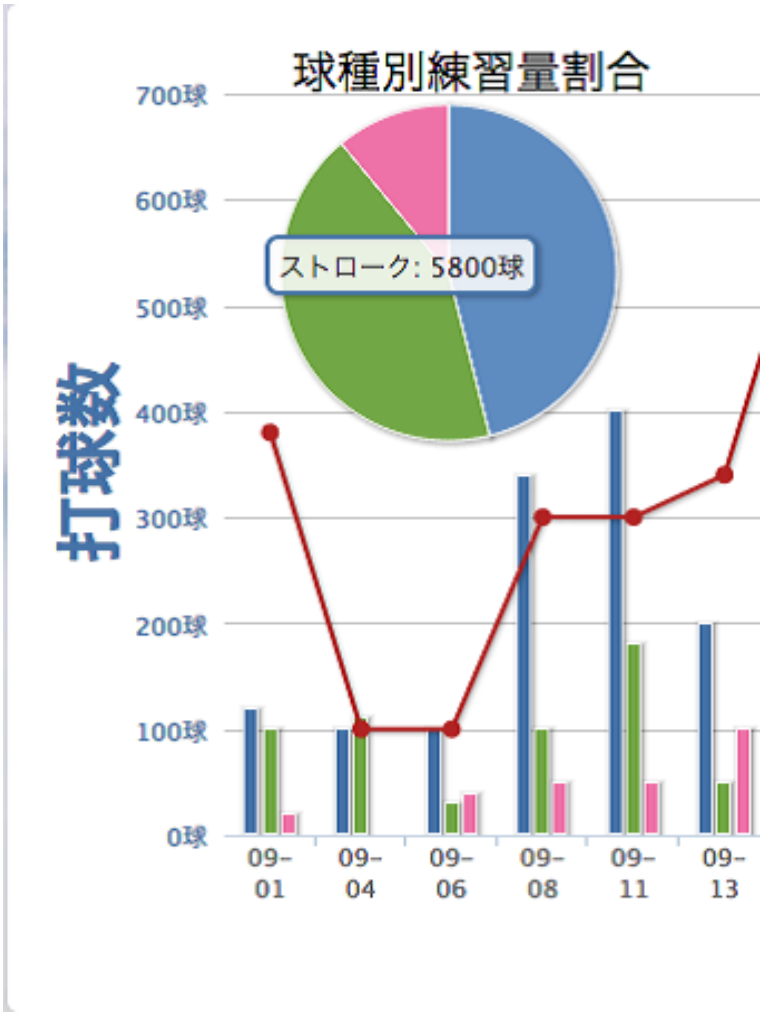

図 2.24: 円グラフにカーソルを合わせる

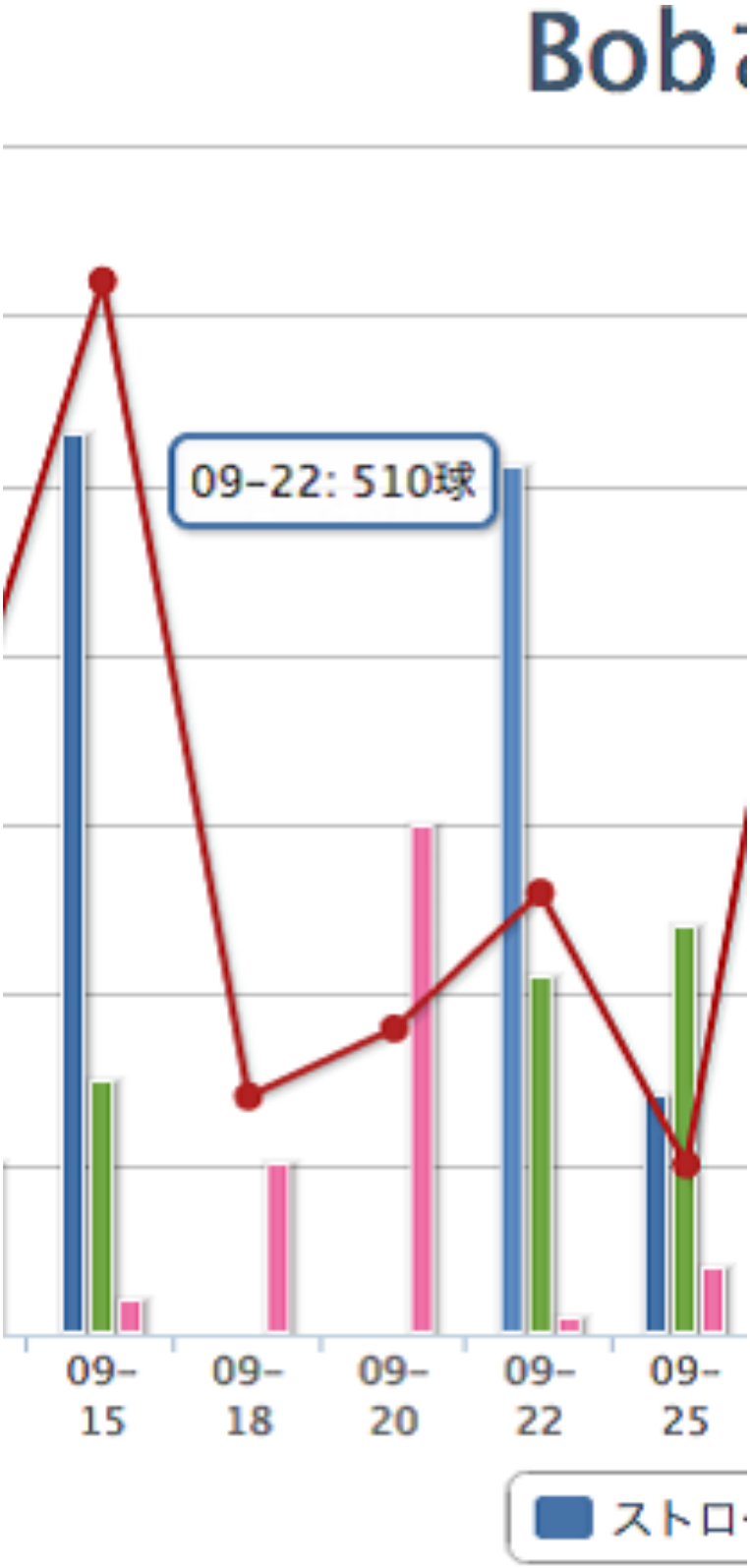

図 2.25: 棒グラフにカーソルを合わせる

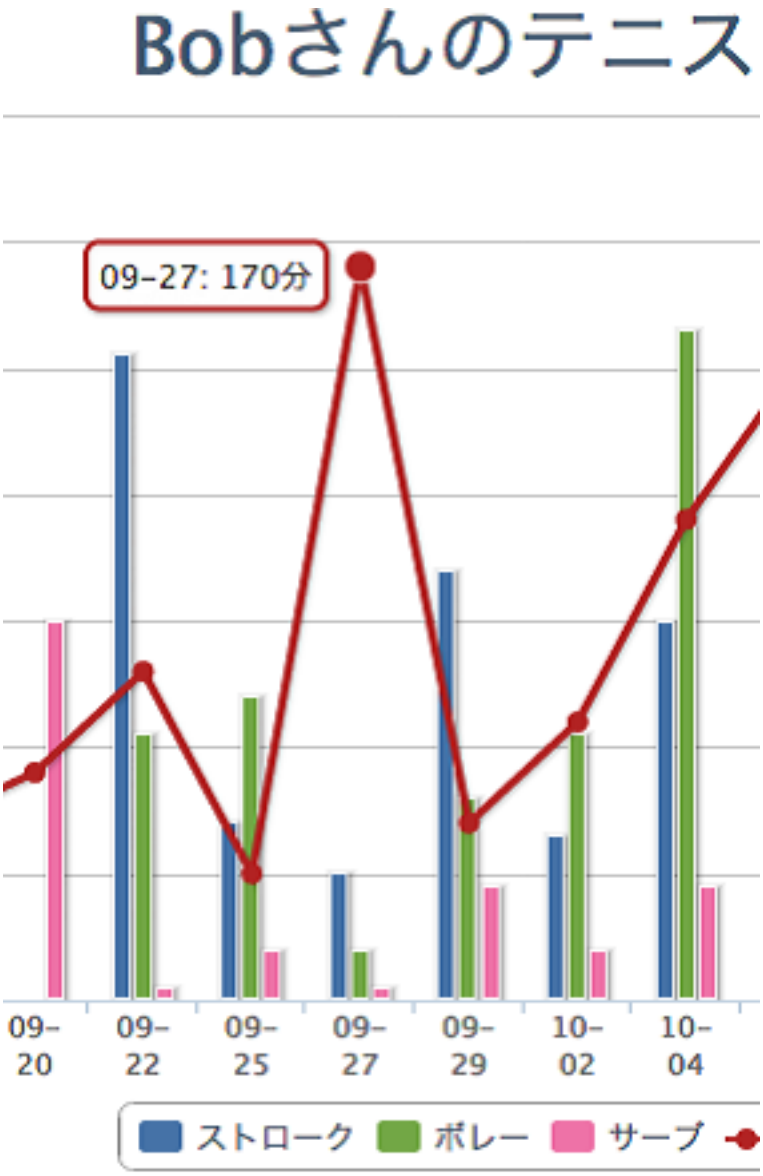

図 2.26: 折れ線グラフにカーソルを合わせる

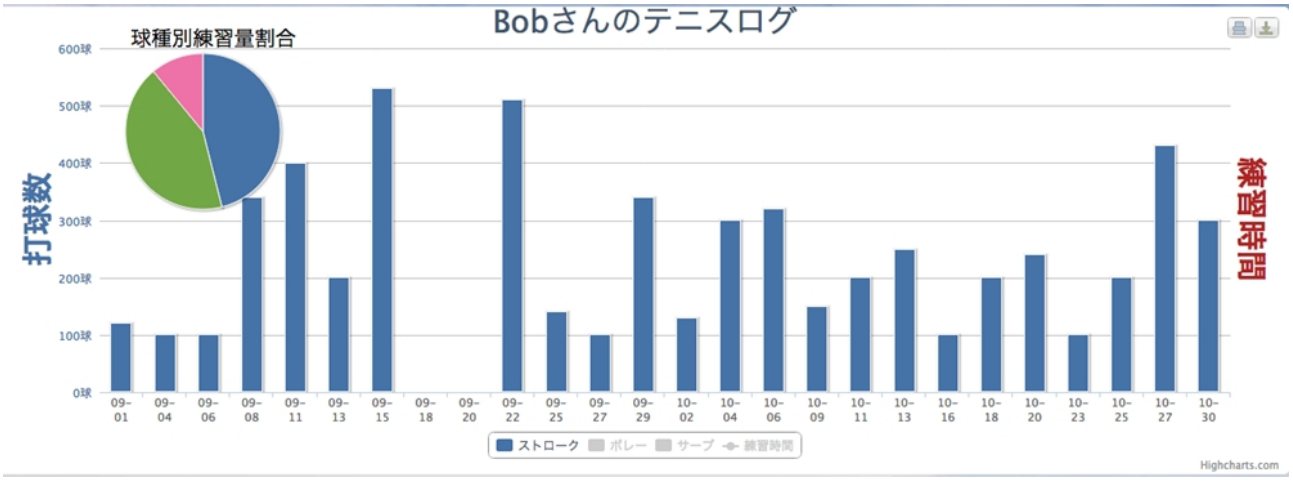

図 2.27: ストローク打球数データのみの表示

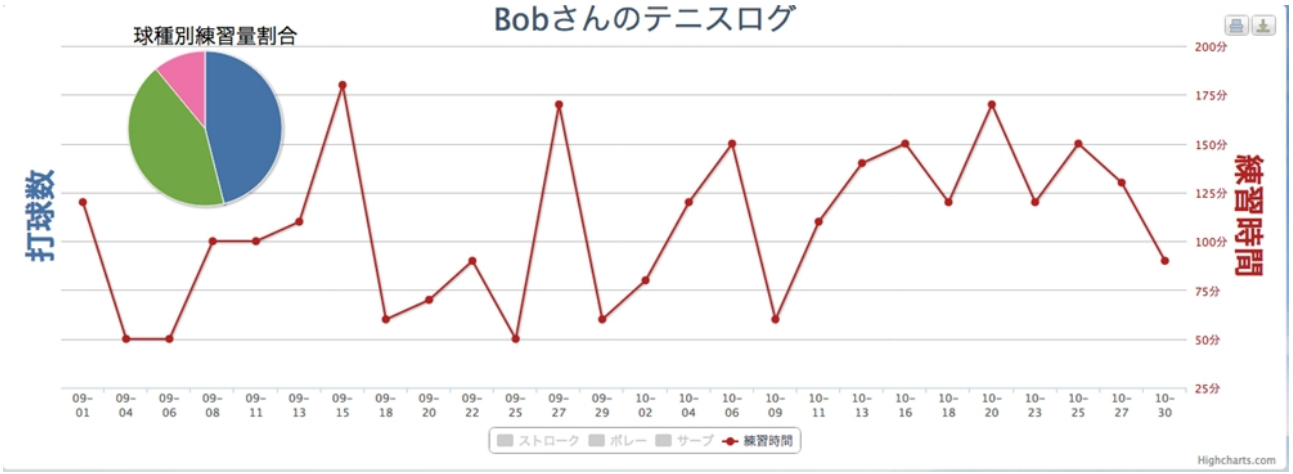

図 2.28: 練習時間データの表示

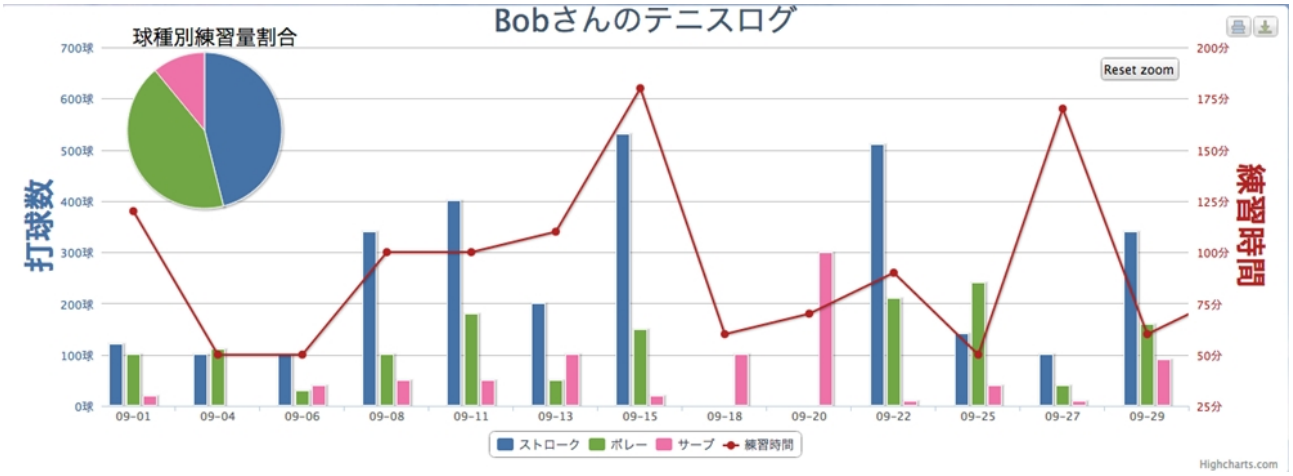

図 2.29: グラフのズーム機能： 1 ヶ月分のみ表示

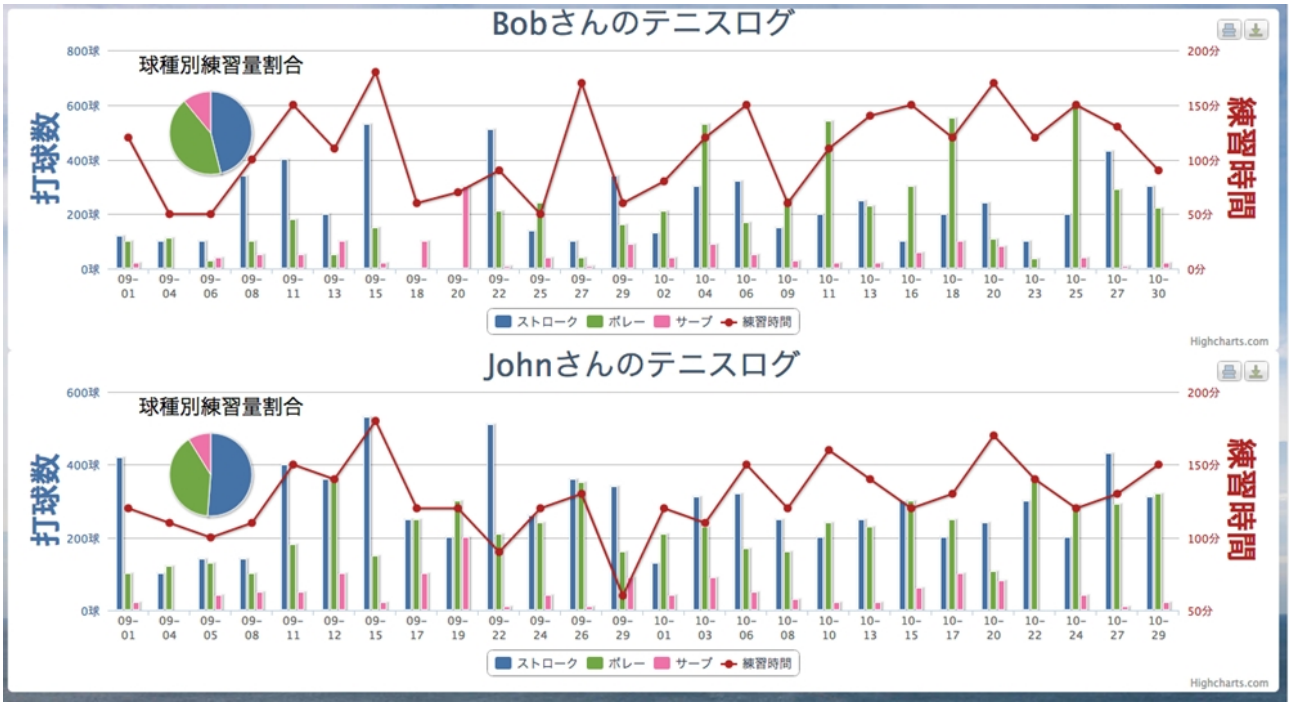

図 2.30: グラフのマルチユーザ表示モード

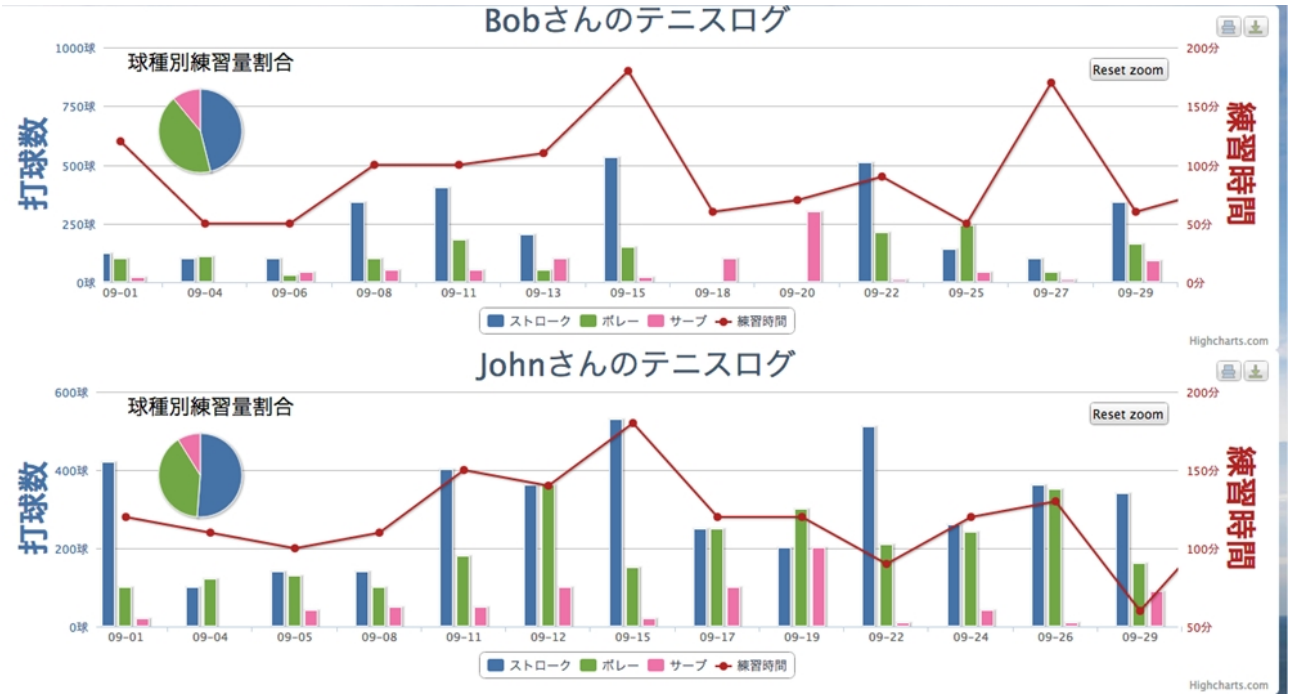

図 2.31: マルチユーザ表示モードで 1ヶ月分のデータを表示

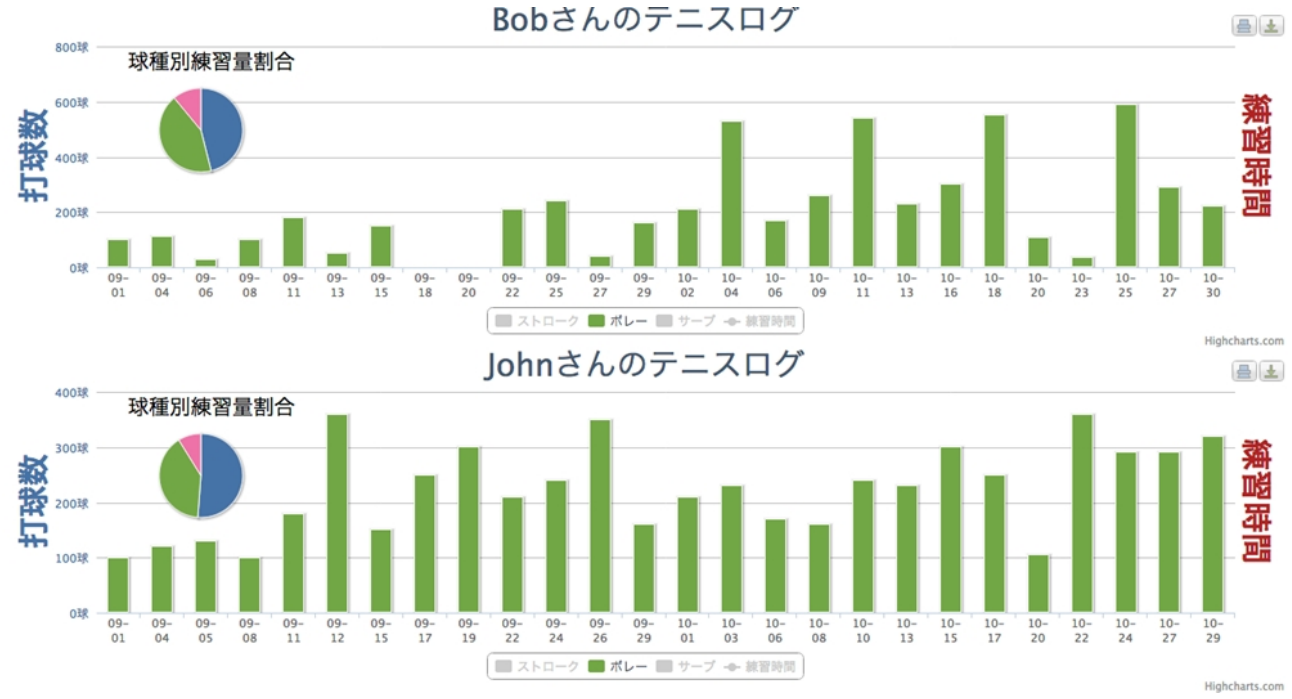

図 2.32: 10 月より Bob のボレー練習量が増加

いかと考えていた。そんなとき John は「テニスログ」の存在を知り、Bob と話し合い、9 月から早速２人で使う事にした。

「テニスログ」を始めて 1ヶ月した 10 月のある日、John はテニスログの球種別表示機能やズーム機能、共有機能などを使い、テニスログのデータ分析を自分なりに行っていた。すると図 2.32 の分析結果画面を見て Bob のボレー練習量の少なさに気づいた。Bob は普段からボレーがうまいため Bob はそれなりにボレー練習をしているのだろうと思っていた。ところが事実は逆であった。John はボレーのうまい Bob がさらにボレーの練習を増やす事で自分らはもっと強くなれると考えた。Bob は図 2.32 の分析結果について John に指摘されると確かに自分のボレー練習量は John と比べ、少ないという事に気づいた。Bob は 10 月よりボレー練習量を増やす事にした。

試合当日朝、Bob と John は２人でテニスログをみながらこの２ヶ月間の練習を振り返っていた。Bob は John に指摘された通り、10 月からボレー練習量を増やしてきた。テニスログでも 10 月よりボレー練習量データが増えており John は試合でその成果が出るのが楽しみになった。

試合まで時間があるので John は前々から気になっていた図 2.31 の 9/18、9/20 において Bob がサーブ練習しかしてない日の事について尋ねた。すると Bob はこの日は約束してた相手がレポート提出に追われ来れなくなり、仕方なくサーブ練習をしたんだと話し始めた。そして Bob はこの日に一人サーブ練習をした事でスピンサーブのコツを掴めたと、サーブ上達のきっかけを教えてくれた。John は Bob の最近のスピンサーブをどこで覚えたか気になっていたがテニスログが話題のきっかけとなり、知る事ができた。

John と Bob は試合前に緊張しがちだがテニスログでお互いの練習量を共有した事で自信がつき、ペアとしての更なる一体感を得られた。テニスログがきっかけで２人はお互いのテニス練習事情を知る事ができ、試合前の緊張も会話でほぐす事ができた。多くの練習量をこなし実力をつけ、リラックスして自信を持ちながら試合に望んだ Bob・John ペアであったが惜しくも準優勝の結果で終わった。しかし２人は次の大会では優勝しようとテニスログと共に次大会での優勝を目指し、練習を始めるのであった。

第3章　テニスログのシステム概要

3.1　システム全体概要

図 3.1 にテニスログのシステム全体概要図を示す。テニスログをログの取得〜活用まで５つのフェーズに分け、テニスログの使用フローを示している。以下にテニスログの 5 つのフェーズを示す。

1. センサデータの取得
2. センサデータのアップロード
3. センサデータの解析
4. パーソナルクラウドへのデータ保存
5. センサデータの活用-グラフ表示-

本システムではセンサデータの取得のためにアンドロイドアプリケーション「テニスロガー」を設計し、Java 言語で実装した。またセンサデータのアップロードからセンサデータの活用を行うため「テニスログホーム」を設計し、PHP 言語で実装した。図 3.2 にテニスログホームのプログラム構成を示す。以下にそれぞれのプログラムの役割を示す。

tennislog_home.php：テニスログホームの起動画面

upload_result.php：センサデータの解析、パーソナルフォルダへの振り分け、保存

hit_read.php：ID 入力の hit_graph.php への受け渡し、エラー処理

hit_graph.php：グラフのシングルユーザ表示

multi_hit_read.php：ID 入力の multi_hit_graph.php への受け渡し、エラー処理

multi_hit_graph.php：グラフのマルチユーザ表示

3.2　センサデータの取得

テニスロガーはアンドロイドプログラミングバイブル [30] のサンプルプログラム「BluetoothEx」をベースに作成した。テニスロガーは３つのクラスから成る。以下に 3 つのクラス

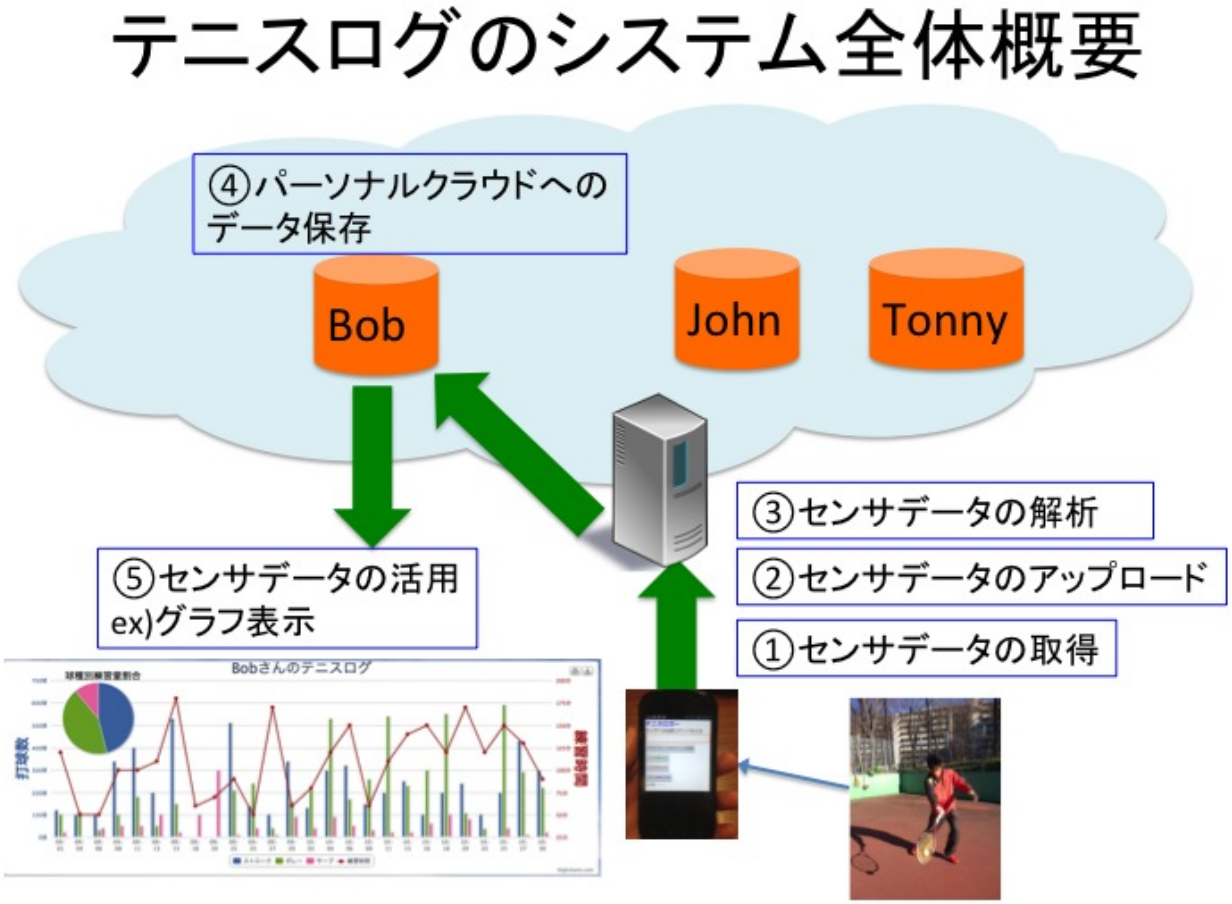

図 3.1: テニスログのシステム全体概要

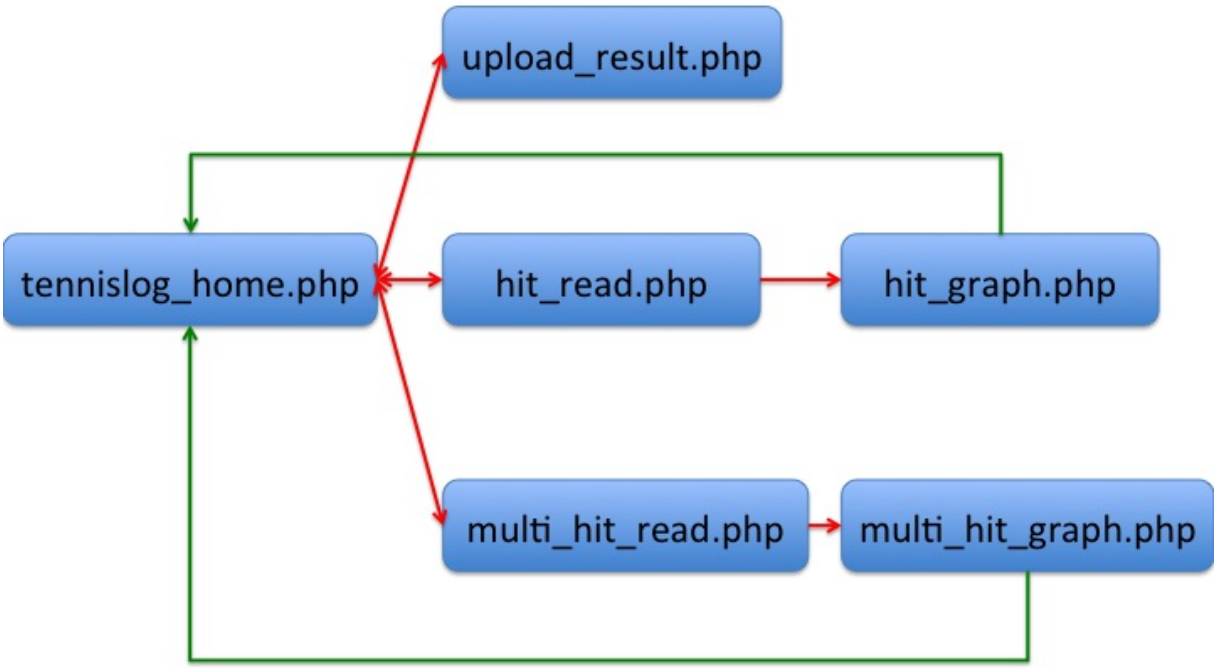

図 3.2: テニスログホームのプログラム構成

それぞれの役割を示す。
・BluetoothEx クラス：センサ操作 (計測開始、停止) コマンドの送信、端末内データ保存処理、UI 処理を行う
・BluetoothChatService クラス：Bluetooth の接続を管理する。Bluetooth の接続待ち、Bluetooth の接続、Bluetooth の切断処理を行う。Bluetooth 接続ではセンサからのデータをストリームで受信し変数に格納、BluetoothEx へのデータ転送を行う
・DeviceListActivity クラス：端末の周囲に存在する Bluetooth 端末を検索する

テニスロガーでは加速度センサより加速度データを取得するためにまず加速度センサとスマートフォンの Bluetooth ペリングを行う。Bluetooth ペアリングとは、2 台の Bluetooth デバイスの接続設定を行う事である。ペアリングを行うことにより Bluetooth 機器は公開している機能へ互いにアクセスできるようになる。

センサ操作コマンドの送信処理

加速度センサはテキスト形式のコマンドを送信する事で計測開始、停止操作を行う事ができる。テニスロガーでは「スタート！」ボタンを押すと計測開始命令テキスト「sens +000003000 5 2 0 +(改行コード)」が加速度センサへ送られる。sens は加速度を意味するテキストである。「sens +000003000 5 2 0 +(改行コード)」は 5 ミリ秒ごとに 2 回サンプリングした平均を出力するという意味である。
「ストップ！」ボタンを押すと計測停止命令テキスト「stop all +(改行コード)」が加速度センサへ送られる。センサは計測開始命令、計測停止命令コマンドを正常に受け付けるといずれもテキスト「OK」を返す。計測開始命令テキストを受け付け、「OK」を返した後、センサは加速度データを送信し始める。図 3.3 に加速度データの中身をスマートフォンで参照した様子を示す。5 ミリ秒ごと 2 回サンプリング平均を出力しているため、5 × 2=10 ミリ秒ごとにセンサデータを受信していることが分かる。図 3.3 の初めのデータを例にとり、加速度データのフォーマットを以下に示す。
「sens,,152544119,1,-8,955+(改行コード)」
「sens」は計測開始命令テキストと同様、加速度を意味するテキストである。「152544119」はタイムスタンプであり、15 時 25 分 44 秒 119 ミリ秒という意味である。「1」「-8」「955」は加速度データであり、左から順に X 軸、Y 軸、Z 軸のデータである。図 3.4 に加速度センサの装着方向図を示す。加速度センサの X 軸、Y 軸、Z 軸の方向は図 3.4 の示す通りである。センサデータは 1 行に 1 レコードが格納され、連続してセンサより送られてくる。

センサデータの受信処理

BluetoothChatService クラスでセンサデータを受信し、ハンドラのターゲットプログラムへの送信メソッド sendToTarget により BluetoothEX へセンサデータを送信する。以下に BluetoothChatService でのセンサデータ受信から BluetoothEx へセンサデータを渡す処理のメイン

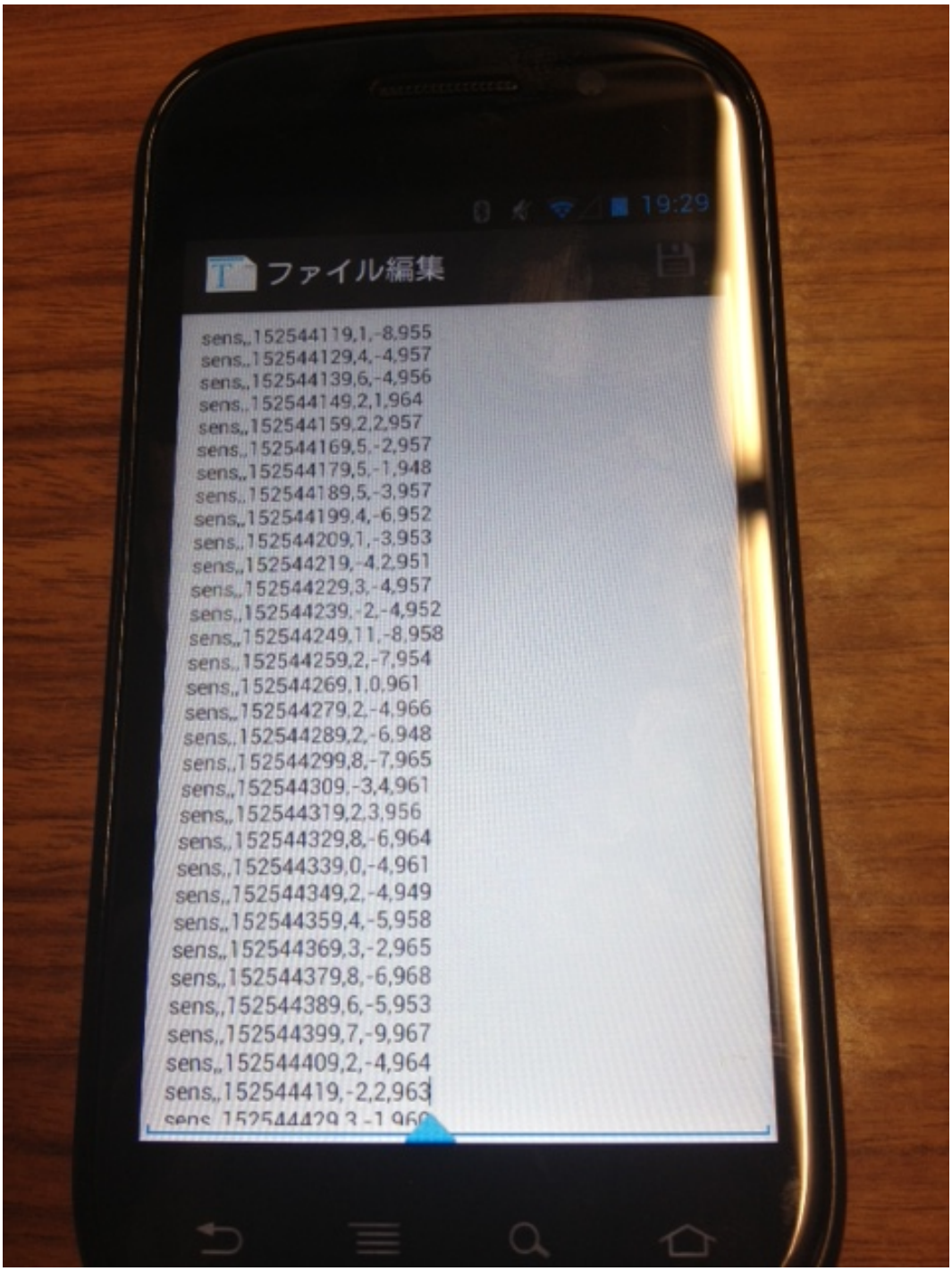

図 3.3: センサデータファイル (CSV 形式) の中身

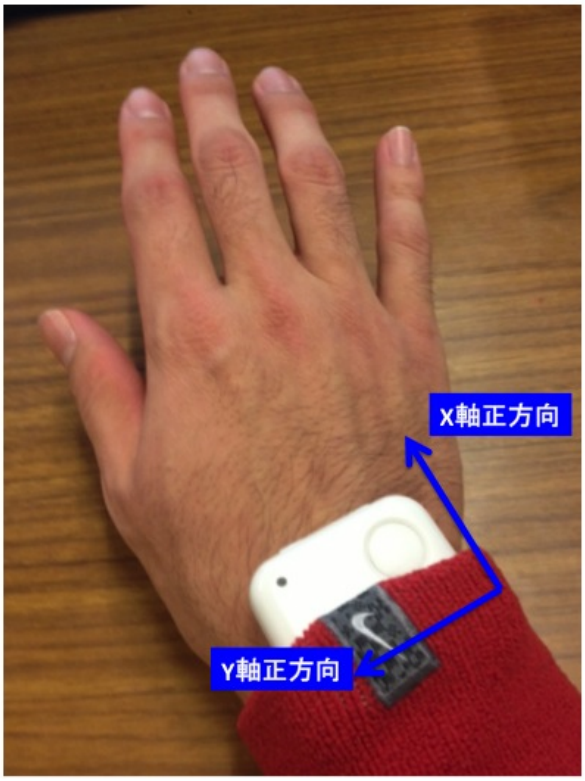

図 3.4: 加速度センサの装着方向

プログラム部分を示す。

```
public synchronized void sleep(long msec){
     try{
     wait(msec);
     }catch(InterruptedException e){}
                   }
    private class ConnectedThread extends Thread {
    private BluetoothSocket socket;
    private InputStream     input=null;
    public ConnectedThread(BluetoothSocket socket) {(1)
    try{
    this.socket=socket;
    this.input =socket.getInputStream();
 }catch(IOException e){}
 }
       public void run(){
         byte[] buf=new byte[16384];(2)
         int buf_bytes;
           while(true){(3)
              try {
                 sleep(2000);(4)
                  buf_bytes=input.read(buf);
                  handler.obtainMessage(BluetoothEx.MSG_READ,
                  buf_bytes,-1,buf).sendToTarget();
                   sleep(2000);(5)
                       }catch(IOException e){
setState(STATE_LISTEN);
break;
}
                   }
              }
          }
```

(1) のコンストラクタにおいて InputStream 変数 input を宣言している。これによりセンサからの受信データは変数 input に格納される。(2) では input からのデータを格納するためのバッファ領域を 16384 バイト確保している。(3) では while 文を無限ループさせる事でバッファ内にたまったデータを常に BluetoothEx に渡す処理を行っている。

(4) の sleep クラスはプログラムの実行を引数で指定した秒数だけ処理を待たせるメソッドである。上記プログラム内 (4) では inputstream に buf で用意した 16386 バイト内で一定量データが溜まるまで待っている。上記プログラム内 (5) では BluetoothEx に buf 内に溜められたデータが全て渡されるまで待っている。sleep クラスを実装する事でセンサデータが受信途中で途切れるなどの破損防止を行っている。

BluetoothEx では BluetoothChatService よりデータを受信すると if 文処理により端末内への保存操作、画面表示テキスト処理を行う。以下に BluetoothEx 内で行うセンサデータの if 文処理のメインプログラム部分を示す。

```
case MSG_READ:
          byte[] readBuf=(byte[])msg.obj;
          if((new String(readBuf,0,msg.arg1)).contains("sens")){
              writefile((new String(readBuf,0,msg.arg1)));
     }
              addText(new String(readBuf,0,msg.arg1));
```

テキストに「sens」が含まれていた場合、writefile クラスにデータが送信される。writefile クラスでは端末内 SD カードへのデータ保存操作を行う。またこの処理がセンサデータよりデータ受信中であることを利用し、同クラス内で画面表示に図 2.12 における「Now Logging ･･･ 」テキストの出力処理を行っている。addText クラスではセンサ接続に関するテキスト表示処理を行っている。図 2.6、図 2.9、図 2.10 に示されるテキスト「センサに接続して下さい」「Now Connecting ･･･ 」「センサ接続完了！「スタート！」ボタンを押すとテニスログが開始されます！」を表示処理している。BluetoothChatService よりセンサ接続の状態を BluetoothEx で受け取り、接続状態に応じた表示処理をしている。

3.3 センサデータのアップロード

図 3.2 において tennislog_home.php より、ID を入力し、センサデータファイルを選択する事でパーソナルクラウドへのファイルアップロードを行う。ID、ファイルを選択し「パーソナルクラウドへ送信」ボタンを押すとパーソナルクラウドへ接続する。本研究ではパーソナルクラウドに NEC ビッグローブクラウドホスティングサービスを利用している。パーソナルクラウドではユーザごとに個人フォルダが作成され、その中に取得したセンサデータファイルが丸ごと格納される。ID 入力について新規に ID 入力を行った場合、パーソナルクラウドに新たに個人フォルダが生成される。既存ユーザの場合は既存の個人フォルダにセンサデータが格納される。ID 入力における新規ユーザと既存ユーザの判別はパーソナルクラウド上の個人フォルダの有無により判定している。

また ID 入力、ファイルアップロード操作についてエラー処理を施している。図 3.5 に ID 入力エラー画面を示す。ID 入力が空であった場合などに表示される。図 3.6 にファイルアップロードエラー画面を示す。ファイルが未選択であったり、アップロード操作がシステムに何らかの理由で認識されなかった場合、csv 形式以外のデータであった場合などに表示される。

テニスログホームでは表記文字に丸みを帯びた「あんず文字」を使用し、装飾している。[31]。

図 3.5: ユーザ ID 入力エラー

図 3.6: ファイルアップロードエラー

3.4 センサデータの解析

以下にセンサデータの解析処理におけるメインプログラム部分を示す。

```
$array_count=0;
$hit_num=0;
$b2_ax=NULL;
$b_ax=NULL;
$countgood=0;
$countbad=0;
while($ar=fgetcsv($fp)){

 if($ar[0]=="sens"
         && $ar[1]==NULL
         &&is_numeric($ar[2])&&is_numeric($ar[3])
         &&is_numeric($ar[4])&&is_numeric($ar[5])){(1)

$t=$ar[2];
$ax=$ar[3];

if(strpos($ar[0],"sens") !== FALSE && $array_count==0){
$start_time=$ar[2];
}

if($b2_ax >= $b_ax && $b_ax < $ax && $b_ax > -16000 && $b_ax< -11000 ){(2)
```

```
$stroke++;
}
else if($b2_ax >= $b_ax && $b_ax < $ax && $b_ax>-11000 &&  $b_ax <-4000 ){(3)
$volley++;
}
else if($b2_ax >= $b_ax && $b_ax < $ax && $ax<-16000){(4)
$serve++;
}

$b2_ax=$b_ax;
$b_ax=$ax;
}else{
continue;
}
$array_count++;
}
$finish_time=$t;
$s_finish_time=substr($finish_time,0,6);(5)
$s_start_time=substr($start_time,0,6);(6)
$time=(strtotime($s_finish_time)-strtotime($s_start_time))/60; (7)
```

センサデータがアップロードされると同時にセンサデータの解析が行われる。解析においても PHP 言語を利用している。本節ではセンサデータの解析アルゴリズムを「打球数検出アルゴリズム」「練習時間検出アルゴリズム」の２つに分けて上記、解析プログラムに触れながら述べる。

3.4.1　打球数検出アルゴリズム

図 3.7〜図 3.10 にストロークのテークバック、インパクト、フォロースルーの様子を示す。テークバックとはラケットを引く動作、インパクトとはボールを打つ瞬間、フォロースルーとは打った後に腕を最後まで振り切る動作の事である。図 3.11 にストロークを 5 球連続で打ったときの加速度変化データを示す。グラフ内赤丸で囲んだ部分がストローク 1 球のインパクト前後の加速度データである。ストロークのインパクト前後では手首の加速度が瞬間的に増加する。そのためグラフ内赤丸で囲んだデータのように瞬間的に加速度値が縦に大きく振れる。図 3.11 では xyz 軸の加速度データを一つのグラフ内で重ねて示している。青色が x 軸、赤色が y 軸、緑色が z 軸である。

図 3.12、図 3.13、図 3.14 に図 3.11 から x 軸、y 軸、z 軸のみのデータをグラフに表した様子を示す。x、y、z 軸のデータのいずれにおいても縦に大きく値が振れている瞬間が見られる。本システムではその中でも x 軸が縦の変化量が最も大きく、打球数検出が行いやすいと考え、

x 軸データのみを打球数検出アルゴリズムに採用した。

上記解析プログラム内において x 軸加速度データは変数、$ax、$b_ax、$b2_ax に格納される。$ax は現在の x 軸データ、$b_ax は 1 つ前の x 軸データ、$b2_ax は 2 つ前の x 軸データである。センサデータは「sens,,152544119,1,-8,955+(改行コード)」のように 1 行 1 レコード形式で送られて来る。解析プログラムでは fgetcsv により 1 行ずつ文字列を読み込み、変数$ar に配列格納する。各配列インデックスに格納されるデータは以下の通りである。

・$ar[0] …「sens」
・$ar[1] … null
・$ar[2] … 15254411(タイムスタンプ)
・$ar[3] … 1(x 軸データ)
・$ar[4] … -8(y 軸データ)
・$ar[5] … 955(z 軸データ)

解析プログラム内 (1) の if 文条件処理では配列格納されたデータが送られてくるレコードのフォーマット通り、正しいデータ型で格納されているかを確認している。$ar[0] は「sens」、$ar[1] は null と常に固定であり、$ar[2]〜$ar[5] では数値文字列が格納されているかを判定している。正しければ打球数検出アルゴリズムを行うための、x 軸データの変数格納、打球数検出処理を行う。

解析プログラム内 (2)(3)(4) においてストローク打球数、ボレー打球数、サーブ打球数の検出処理を行っている。打球数検出には時系列で連続した 3 つの x 軸データを使う。図 3.12 より、インパクトの瞬間 x 軸データは下に凸な変化グラフとなる。この凸時のデータと凸前後の x 軸データが一定値の間で検出された場合、1 回の打球数とカウントする。解析プログラム内では-16000 から-11000 の間において凸部分が検出された場合 1 回の打球数とカウントしている。なお値の単位は重力加速度の 1000 分の 1 である mG である。同様にしてボレー、サーブの打球数検出処理を行っている。解析プログラム内ではボレーは-11000 から-4000 の間の凸部分データ、サーブは-16000 以下において凸部分が検出された場合に 1 回の打球数とカウントしている。

図 3.15〜図 3.17 にボレーのテークバック、インパクト、フォロースルーの様子、図 3.19〜図 3.22 にサーブのテークバック、インパクト、フォロースルーの様子を示す。ボレー、サーブにおいてもテークバック、インパクト、フォロースルーの意味はストロークと同じである。ボレー、サーブにおいてもインパクト前後で手首の動きが一瞬速くなるため、加速度データは縦に大きく変化する。図 3.18 にボレーの x 軸加速度データが縦に大きく変化している様子を示す。図 3.23 にサーブの x 軸加速度データが縦に大きく変化している様子を示す。図 3.18、図 3.23 はどちらも 5 球連続で打っており、画像内の赤丸が 1 球分のインパクト前後の x 軸加速度データである。

3.4.2　練習時間検出アルゴリズム

加速度データには変数$ar[2] にタイムスタンプが含まれている。タイムスタンプを利用して「練習時間」データを算出する。解析プログラム内では初めのタイムスタンプを練習時間

図 3.7: ストローク：テークバック

図 3.8: ストローク：インパクト前

の開始時間として、変数$start_time に格納している。また while 文による fgetcsv のデータ読み込みが終了した直後、$ar[2] に格納されているタイムスタンプを練習終了時間として変数$finish_time に格納している。変数$start_time と$finish_time はミリ秒まで計っている。そのため解析プログラム内 (5)(6) においてミリ秒以下は切り捨てる処理を行っている。(7) では$start_time と$finish_time を UNIX タイムスタンプで時刻単位を統一した後、(終了時刻-開始時刻) の引き算処理を行い、練習時間を算出している。

3.5 パーソナルクラウドへのデータ保存

図 3.2 において upload_result.php でパーソナルクラウドへのデータ保存操作を行う。ファイル名は「ユーザ ID ＋ファイルアップロード日時」が自動的に割り当てられ、保存される。また生データを解析し、MySQLDB に解析データを保存する。

3.5.1 生データの保存

ユーザごとに一つのフォルダ (パーソナルフォルダ) が生成される。アップロードされた CSV 形式のセンサ生データは全てこのパーソナルフォルダに格納される。テニスログホームの起動画面においてユーザ ID が新規か既存かを判別する際に、パーソナルフォルダの存在により

図 3.9: ストローク：インパクト

図 3.10: ストローク：フォロースルー

図 3.11: ストローク：XYZ 軸加速度データ

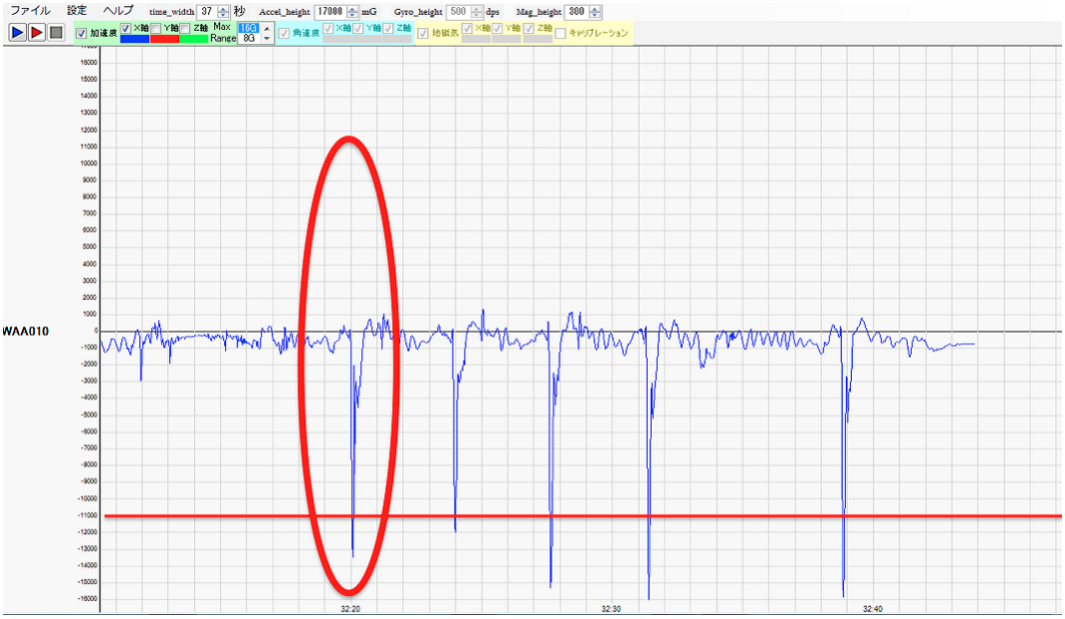

図 3.12: ストローク：X 軸加速度データ

図 3.13: ストローク：Y 軸加速度データ

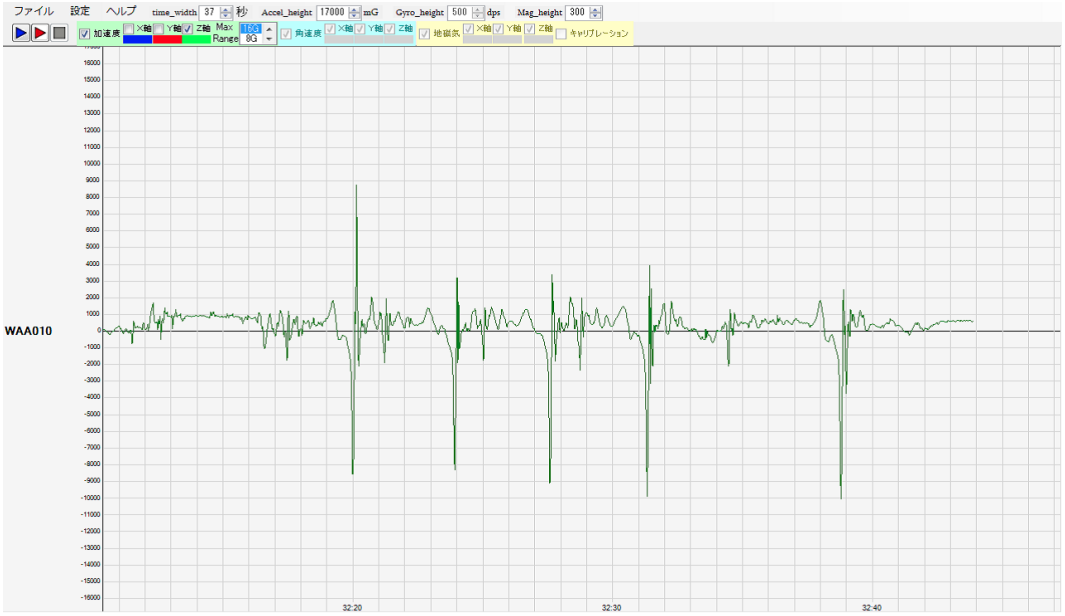

図 3.14: ストローク：Z 軸加速度データ

図 3.15: ボレー：テークバック

図 3.16: ボレー：インパクト

図 3.17: ボレー:フォロースルー

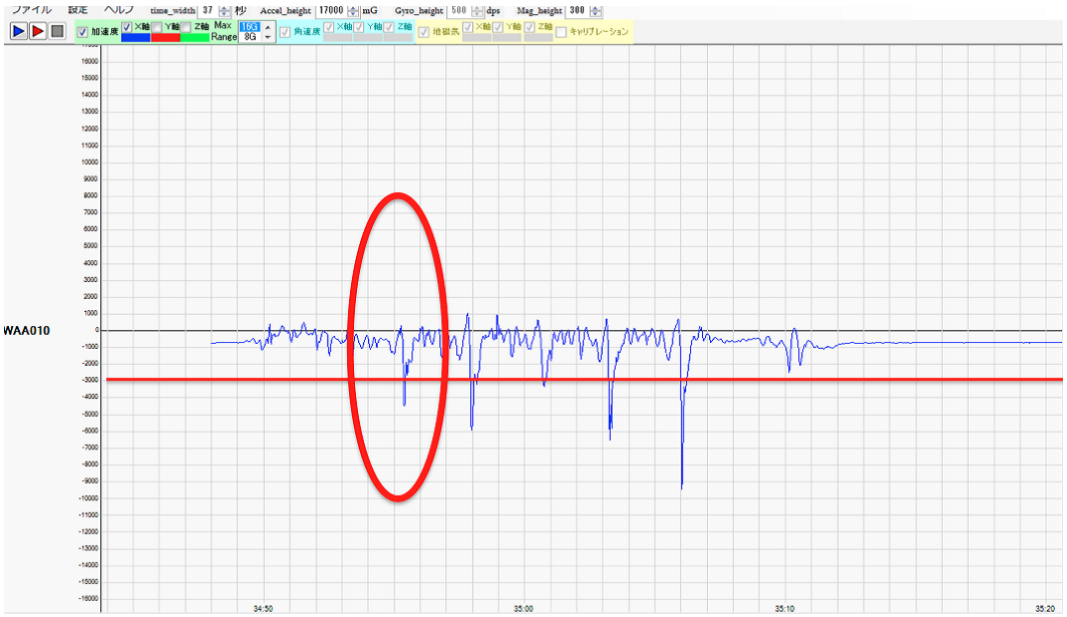

図 3.18: ボレー：X 軸加速度データ

図 3.19: サーブ：テークバック

図 3.20: サーブ：インパクト前

図 3.21: サーブ：インパクト直前

図 3.22: サーブ：フォロースルー

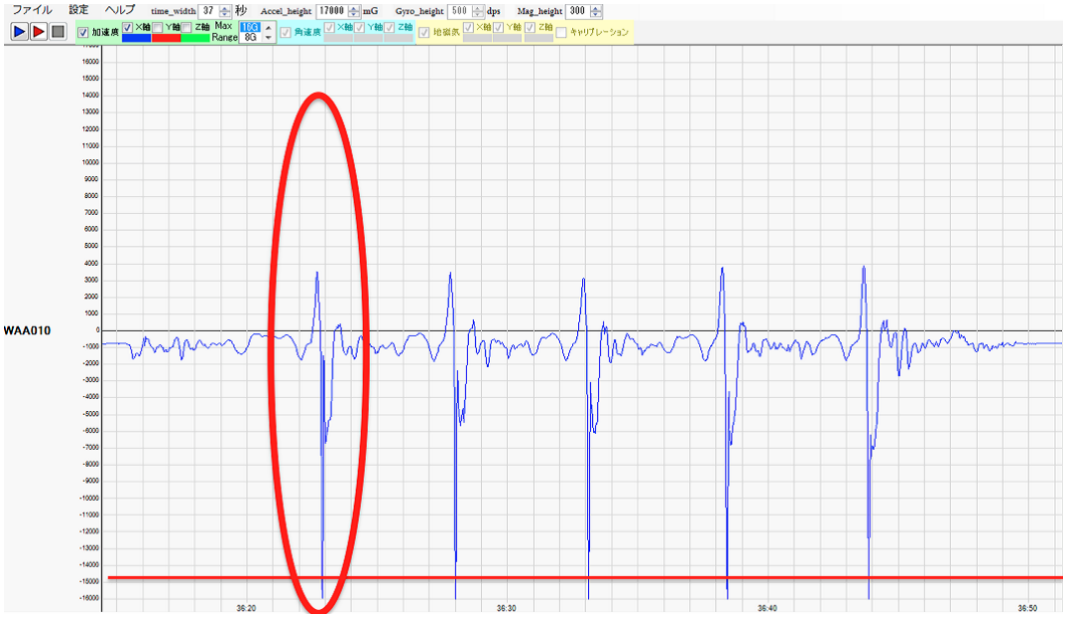

図 3.23: サーブ：X 軸加速度データ

判別する。判別には file_exists 関数を使い、引数にサーバ内でのパーソナルフォルダへの絶対パスを指定する。

図 3.24 にユーザごとのパーソナルフォルダ、 図 3.25 にパーソナルフォルダ内のセンサ生データファイルの一覧を示す。図 3.24 内ではユーザ「John」「ibuki」「ibu2」「ibu」「Bob」のパーソナルフォルダが表示されている。図 3.25 ではユーザ「ibu」内のパーソナルフォルダの中身を示している。ファイル名の先頭に「ibu-」とついており、日時ファイル名で送られてくるセンサデータファイル名にユーザ ID が付加されている事が分かる。

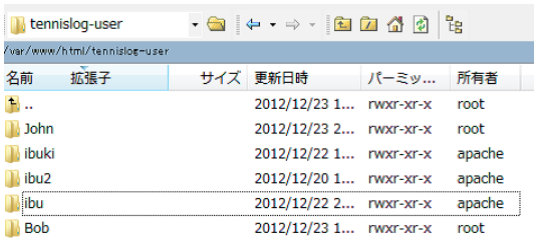

図 3.24: ユーザごとのパーソナルフォルダ

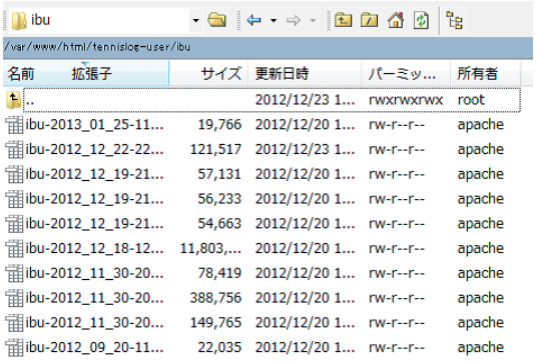

図 3.25: パーソナルフォルダ内のセンサ生データファイル

3.5.2 解析データの保存

解析データ (ストローク打球数、ボレー打球数、サーブ打球数、練習時間) はパーソナルクラウド上の MySQLSB に保存される。MySQLDB 接続にはホスト名、接続ユーザ、パスワード、データベース名を指定する。MySQLDB ではユーザごとにテーブルが作成され、テーブルには以下のフィールドが用意されている。図 3.26 に解析データが格納された MySQLDB テーブルを示す。

id: データの挿入順に自動的に割り振られる。データの個数などが分かる。

date: センサデータの取得日時を表示。グラフ表示において日付順にソートする際、使用する。

time: テニスの練習時間。単位は分。

filename: CSV 形式のセンサ生データのファイル名。「ユーザ名+センサデータの取得日時.CSV」で構成されている。

stroke: ストローク打球数。単位は球。

volley: ボレー打球数。単位は球。

serve: サーブ打球数。単位は球。

			id	date	time	filename	stroke	volley	serve
☐	🖊	✕	1	2012-09-01	120	Bob-2012-09-01.csv	420	100	20
☐	🖊	✕	2	2012-09-08	110	Bob-2012-09-08.csv	340	100	50
☐	🖊	✕	3	2012-09-15	180	Bob-2012-09-15.csv	530	150	20
☐	🖊	✕	4	2012-09-22	90	Bob-2012-09-22.csv	510	210	10
☐	🖊	✕	5	2012-09-29	60	Bob-2012-09-29.csv	340	160	90
☐	🖊	✕	6	2012-10-06	150	Bob-2012-10-06.csv	320	170	50
☐	🖊	✕	7	2012-10-13	140	Bob-2012-10-13.csv	250	230	20
☐	🖊	✕	8	2012-10-20	170	Bob-2012-10-20.csv	240	106	80
☐	🖊	✕	9	2012-10-27	130	Bob-2012-10-27.csv	430	290	10
☐	🖊	✕	10	2012-09-04	110	Bob-2012-09-04.csv	200	120	0

図 3.26: 解析データが保存されてる MySQLDB テーブル

ユーザテーブル名とユーザ ID は一致している。新規ユーザ ID の場合、新規パーソナルフォルダが作成すると同時に新規テーブルを作成する SQL クエリを送信し、新規ユーザテーブルを作成する。以下に新規ユーザテーブルを作成する際の SQL クエリを示す。$username はテニスログホームで入力されたユーザ ID である。values には解析データが格納されている変数を指定している。

```
$sql_insert="insert into $username(date,time,filename,stroke,volley,serve)

values('$file_date','$time','$filename','$stroke','$volley','$serve')";
```

3.6 センサデータの活用-グラフ表示-

図 3.2 において hit_graph.php によりグラフのシングルユーザ表示を行う。multi_hit_graph.php によりグラフのマルチユーザ表示を行う。グラフ表示には Highcharts ライブラリ [32] を利用した。Highcharts とはグラフを作成できる JavaScript ライブラリである。hit_graph.php、multi_hit_graph.php ではプログラム内で highcharts.js、exporting.js の 2 つのグラフライブラリを参照している。

横軸、縦軸データについて JavaScript 内で PHP 言語を記述し、MySQLDB を呼び出す事でデータ挿入を行っている。以下に、横軸データである日付データを挿入するプログラム部分を示す。日付については MySQLDB よりデータを取り出す際、ソート表示処理を行っている。

```
xAxis: [{
                categories: <?php
        $host = 'tennislog';
```

```
$user ='root';
$pass ='********';
$dbname = 'tennis';
$username=$_SESSION["username"];
$conn = mysql_connect($host, $user, $pass) or die("接続エラー");
mysql_select_db($dbname) or die("接続エラー");(1)
//日付順にソート
$sql="SELECT * FROM $username ORDER BY date DESC ";(2)
$res = mysql_query($sql, $conn) or die("データ抽出エラー");
$str2=NULL;
while ($row= mysql_fetch_array($res,MYSQL_ASSOC)) {(3)
$row_sub=substr($row[date],5,5);(4)
$row_x="'$row_sub'";
$str2="$row_x".","."$str2";(5)
}
                echo"[".$str2."]";(6)
               ?>
```

横軸のデータ挿入においてまず php の開始タグと終了タグでプログラムを囲み、PHP 言語の記述を JavaScript 構文内で行っている。$host、$user、$pass、$dbname を指定し MySQLDB 接続を (1) で行っている。(2) では図 3.26 において date フィールドを日付順にソートしてダウンロードする SQL クエリである。(3) では while 文により 1 行 1 レコードデータを変数配列に格納している。データの取り出しは MySQL のフィールド名を配列インデックスに指定する事でアクセスできる。上記プログラムでは日付データのダウンロードを行っているので「date」をしている。(4) では日付のうち、月日だけのデータを取り出すため、substr 関数により年数が入っているはじめの 4 バイトを切り取り、月日だけを変数$row_sub に格納している。(5) では日付データをカンマ区切りで連続した一つのストリングへと変換処理し、$str2 に格納している。既存のデータに新たなデータをストリング初めに追加している。以下に (6) により出力されるストリング例を示す。

「09-01,09-04,09-06 ･･･ 」

縦軸データにおいても同様の処理でデータ挿入を行っている。縦軸データの挿入処理では上記プログラム内の (4) においてインデックスを MySQLDB のフィールド「stroke」「volley」「serve」「time」に変更する。

エラー処理

図 3.2 において hit_read.php、multi_hit_read.php でユーザ ID 入力のエラー処理を行っている。図 2.16 において ID 入力を間違えた場合などのエラー警告画面を図 3.27、図 3.28 に示す。図

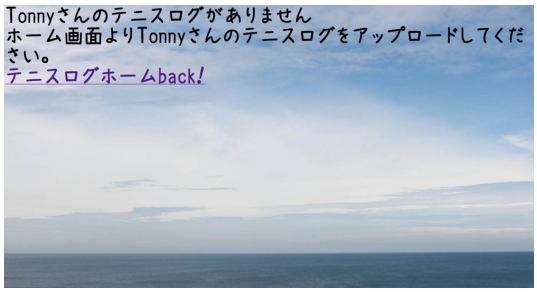

図 3.27: シングルユーザ表示での ID 入力エラー

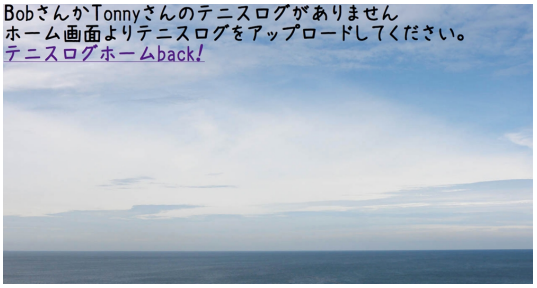

図 3.28: マルチユーザ表示での ID 入力エラー

3.27、図 3.28 では「ユーザ ID さんのテニスログがありません」と表示し、「ホーム画面より
テニスログをアップロードしてください」と表示させている。

図 2.16 で入力された ID はセッション変数に保存され上記プログラム内の$username=$_SESSION[”username”];
により別プログラムでの変数$username に格納される。これにより MySQLDB へのユーザテー
ブルへのアクセスが可能となる。

第4章　関連研究

本章ではパーソナルクラウドを用いたライフログシステムの関連研究として「ライフログを共有する Life Networkig Service」[33]、「Smart Makeup System」[34]、「Life Log Media 」[23]、「レシートログ」[16] の 4 つのシステムを挙げる。テニスにおける関連研究としてテニススウィングを解析した研究 [35] を挙げる。またコンピュータ技術を使ったテニス解析研究として IBM 社の SlamTracker[36]、北原ら [37] の画像処理を用いたテニス競技のデータ自動記録、植村ら [38] のテニス試合におけるポイントごとの動画シーン抽出を行う研究を紹介する。

4.1　ライフログを共有する Life Networking Service

Life Networking Service(LNS) では家族や家電、ヘルスケア機器から得られたデータをパーソナルクラウドに蓄積させ、管理する。LNS は SNS と連携させ、マイクロブログ形式で離れて暮らす家族に生活や健康状態を通知させる。LNS では家族など自分に身近な小規模グループの人とのライフログ共有を行う。パーソナルクラウド上にデータを蓄積させ、ログ共有を行うという点で本研究と関連している。LNS では Twitter API を用いて LNS からの情報をテキスト表示させているが、本研究ではログの表示にグラフ形式を採用しており、ライフログ共有の際のログ表示形式が異なっている。

4.2　Smart Makeup System

Smart Makeup System(SMS) では女性の化粧ログをパーソナルクラウドに記録し、他者と共有するシステムである。SMS では化粧のバリエーション増加を目的としている。SMS で取得するログデータは化粧後の顔写真と使用した化粧品情報である。これらログデータを取得するため SMS では Smart Makeup Logger(SML) により、Web カメラ、RFID リーダ、バーコードリーダを使って顔写真、化粧品情報を取得している。取得したログの保存、活用するために Smart Makeup Communicator(SMC) では、Web ブラウザで化粧ログの検索や共有を行っている。化粧ログの共有により化粧のバリエーションが増えると述べている。新たな知見を得るために他人とログ共有を行うという点で本研究と似ている。本研究とはログの取得に用いるデバイスが異なっている。またログの表示形式が SMS では写真、本研究ではグラフであるという点が異なっている。

4.3 Life Log Media

Life Log Media(LLM) では生活の情報をセンサを用いて収集し、サーバに蓄積させ、解析し、ログ情報をブラウザで閲覧するシステムである。ログの収集に用いるセンサ機器類は加速度センサ、マイクロホン、HMD、カメラなどである。これらセンサ機器類を PDA などの携帯情報端末と無線通信させる。携帯情報端末は一時的なログ保存機能を持ち、ネットワークにつながった際にサーバへのアップロード作業を行う。ログ解析はサーバ上で行う。本研究とはセンサでのログ取得、携帯情報端末を用いたアップロード、サーバで解析を行うというログの取得〜活用の一連の動作が似ている。LLM では携帯情報端末に PDA を使用し、サーバ機能は Windows XP 上に構築しているが、本研究では携帯情報端末にスマートフォンを使用し、サーバはビッグローブクラウドホスティングを利用している点で異なる。

4.4 レシートログ

徳永らのレシートログではユーザの購買履歴をレシートから読み取り、DB に蓄積させる。DB にユーザテーブル、レシートテーブルを設け、ユーザ ID を主キーとしてログ検索機能を実装している。ユーザテーブルを設け、ユーザ ID を用いてログ検索機能を実装している点で本研究と似ている。またレシートログを SNS で共有する機能を検討している。ログの複数人共有を検討している点において本研究と似ている。徳永らはログの共有機能を今後の課題としているが、本研究ではパーソナルクラウドにより、ログの複数人共有機能を実装し、解決しているという点で異なる。

4.5 テニススウィングの解析と定量評価

浅野らは 2 台のカメラを用いてステレオ法によりボール、ラケットの 3 次元位置を検出する研究を行った。ラケットのスイング軌道とボールの位置を解析し、初心者と上級者の軌道の違いなどの定量的評価を行っている。テニススウィングの解析を行っているという点で本研究と関連がある。本研究のシステムではスイングを解析し、テニス練習量を蓄積させ、蓄積データの解析を行っており、この点が異なっている。

4.6 SlamTracker

IBM 社はテニスの試合をリアルタイムで解析し、Web 上で閲覧できるサービス「SlamTracker」を提供している。SlamTrcker ではカメラやセンサを用いてテニスコート上から選手の動き、ボールの動きなどを取得する。解析から得られるデータはスコアやラリー数、サービスエースの本数、確率、ミスの確率、リターンでのヒッティングポイントなど多岐にわたる。さらに選手の過去の蓄積データからや勝利、敗北要因を解析したり、サービスエースの合計本数、ミスの確率から今日の選手の調子などを解析する事ができる。

4.7　画像処理を用いたテニス競技のデータ自動記録・分析手法

北原らは市販のカメラ 2 台を用いてテニスにおけるプレーヤの動き、ボールの軌跡、バウンド地点、打点などを自動で記録し、ユーザに提示する手法を提案している。提案手法では実際にフォアハンドストローク、バックハンドストロークを打った割合、打ったコース (フォアサイド、バックサイド) の判定、記録を行っている。検出法はあらかじめ誰もいないテニスコートを背景画像として用意し、人とボールが移った入力画像との差分により人とボールの位置検出を行う。現在のボール座標と一つ前のフレームのボール座標を直線で結んだものをベクトルして表し、ベクトル方向が急激に変化した点を、打点、バウンド地点として推定している。打点は人の真横で発生すると仮定して、打点と人の位置からフォアハンド、バックハンドを判定する。

4.8　テニス試合におけるポイントごとの動画シーン抽出システム

植村らは一度見たテニス試合動画から目的のシーンを容易に検索できる手法を提案している。提案手法ではテニスの試合動画をセット、得点ごとに分割し、簡易図化を行う。簡易図化ではコートとボールの軌跡を線で表している。これにより、ユーザが目的とするシーンが探しやすくなると述べている。ポイントごとの動画抽出処理では、前フレームとの RGB 各平均値の差が閾値以上である事、さらにコートが検出されてる事を利用する。テニスコートが検出されるフレームがあまりに短い場合はブレーが行われなかったとみなして、そのフレームをカットする。このようにしてプレー中と思われるフレームのみを抽出し、人の動きとボールの動きからプレーを簡易図化した図を作成する。簡易図化した図をクリックするとそのプレー動画を視聴する事が出来る。

第5章　まとめと今後の課題

本研究ではライフログの取得、蓄積、活用の 3 フェーズそれぞれの課題を考えるためにパーソナルクラウドを用いたテニス練習記録の蓄積、共有支援ツール「テニスログ」の設計、実装を行った。以下に 3 フェーズそれぞれについてのまとめと今後の課題を述べる。また最後にパーソナルクラウドについての考察を述べる。

5.1　ライフログ取得フェーズのまとめと今後の課題

ライフログ取得フェーズではテニス練習において加速度センサとスマートフォンを用いて練習量データを取得するアンドロイドアプリケーション「テニスロガー」を設計、実装した。本研究では「テニスロガー」によって加速度センサからログの自動取得が行える事を確認した。

今後の課題として、ログを自動的にサーバへアップロードする仕組みを設計するという事が考えられる。自動アップロードの仕組みとして 1. 中継端末である携帯情報端末がセンサより取得したログを自動的にサーバへアップロードする、2. センサから直接、ログをサーバへアップロードするデバイスを使用するなどの手法が考えられる。

5.2　ライフログ蓄積、活用フェーズのまとめと今後の課題

ライフログ蓄積、活用フェーズでは加速度センサから取得したログをパーソナルクラウド上に蓄積させ、解析結果をグラフ表示させた。本研究では加速度センサから取得したログをパーソナルクラウド上に蓄積させた事で携帯情報端末、デスクトップ PC など、様々なデバイスから個人のライフログデータにアクセスできるようになった事を確認した。様々なデバイスからログにアクセスできるためログの複数人共有が可能になったと考えられる。今後の課題として、ログは共有する事で自分視点以外の新たな知見を得られるか、自分視点以外の新たな知見を得るために追加すべき機能は何か、などログ共有から知見を得るための仕組みを検討する必要があると考えられる。

5.3　考察

最後にパーソナルクラウドがライフログシステムで解決した問題とパーソナルクラウドへのログアップロード方式についてのまとめを述べる。

本研究ではライフログシステムにパーソナルクラウドの概念、技術を取り入れる事で、ライフログにおけるストレージ、解析処理の問題を解決した。これまではクラウドストレージの容量制限や通信速度が遅いという問題があり、パーソナルクラウド上でライフログの蓄積、解析処理を行う事は難しかった。しかし、近年の通信高速化、クラウドストレージの大容量化により、パーソナルクラウド上にログ蓄積が行えるようになった。

携帯情報端末を用いてパーソナルクラウドへライフログの取得、アップロード作業を行う場合、次の２通りの方法が考えられる。

1 : 携帯情報端末で取得した生データを解析し、解析結果をサーバにアップロードする

2 : 携帯情報端末で取得した生データをサーバにアップロードし、解析処理を行う

１のメリットはアップロードする際のデータ量が少なく、通信が高速化しやすいという事である。デメリットは携帯情報端末ではデータ処理に時間がかかるという事である。

２のメリットはデータ処理をサーバ側で高速に行えるという事である。デメリットはアップロードする際のデータ容量が大きく、通信が遅くなりやすいという事である。

テニスログでは２の方式をとった。生データをクラウド側で処理した事で、処理の高速化、利用者間の比較など、高度な処理を行えた。将来のライフログシステムにおいては取得するデータ量、携帯情報端末の性能、通信環境により 1、2 の方式を使い分ける事が望ましいと考えられる。

謝辞

本論文を執筆するに当たり、指導教員である田中二郎先生をはじめ、志築文太郎先生、三末和男先生、高橋伸先生には丁寧なご指導、有益なアドバイスを頂き、心から感謝を申し上げます。

また本研究を行うに当たり数多くの有益なアドバイスを下さった NEC ビッグローブの神場知成様、村上隆浩様に心より深く感謝を申し上げます。

そしてインタラクティブプログラミング研究室の皆様には日頃から研究に関し、様々な意見、アドバイス、励ましをいただきました。ここに厚くお礼申し上げます。

最後に私の大学生活を支えて下さった、家族、友人、お世話になったすべての方にこの場を借りて深くお礼申し上げます。ありがとうございました。

参考文献

[1] 平田 恭二, 塩梅 健治, 粟冠 徳幸, 加藤 剛史, 神場 知成, パーソナルクラウドの実現に向けて, NEC 技報, Vol.62,No.4, pp88-91(2009).

[2] Jens-Sonke Vockler, Gideon Juve, Ewa Deelman, Mats Rynge, Bruce Berriman, Experiences using cloud computing for a scientific workflow application, ScienceCloud pp.15-24(2011).

[3] Kenneth Nagin, David Hadas, Zvi Dubitzky, Alex Glikson, Irit Loy, Benny Rochwerger, Liran Schour, Inter-cloud mobility of virtual machines, SYSTOR Article No.3(2011).

[4] Ye Wen, Xiaokang Yang, Yi Xu, Cloud-computing-based framework for multi-camera topology inference in smart city sensing system, MCMC'10, pp.65-70(2010).

[5] Facebook:https://www.facebook.com/home.php.

[6] Evernote:https://evernote.com/intl/jp/.

[7] Masayuki Ono, Kunihiro Nishimura, Tomohiro Tanikawa, Michitaka Hirose, Structuring of Lifelog Captured with Multiple Sensors by Using Neural Network, Proceedings of the 9th ACM SIGGRAPH Conference on Virtual-Reality Continuum and its Applications in Industry on VRCAI'10, pp.111-118(2010).

[8] Joshua McVeigh-Schultz, Jennifer Stein, Jacob Boyle, Emily Duff, Jeff Watson, Avimaan Syam, Amanda Tasse, Simon Wiscombe, Scott Fisher, Vehicular lifelogging: new contexts and methodologies for human-car interaction, CHI EA '12 pp.-221-230(2012).

[9] Pil Ho, Fausto Giunchiglia, Lifelog event management: crowd research case study, Proceedings of the 2011 joint ACM workshop on Modeling and representing events on J-MRE'11, pp.43-48(2011).

[10] 大久保成晃, 菊池 浩明, 各種センサを統合した行動状態推定機能を持つライフログシステムの Android への実装と評価, 情報処理学会第 74 回全国大会, pp.351-352(2012).

[11] Daragh Byrne, Aisling Kelliher, Gareth J.F. Jones, Life editing: third-party perspectives on lifelog content, CHI '11, pp.1501-1510(2011).

[12] Sian E. Lindley, Dave Randall, Wes Sharrock, Maxine Glancy, Nicola Smyth, Richard Harper, Narrative, Memory and Practice: Tensions and Choices in the Use of a Digital Artefact, BCS-HCI 09', pp.1-9(2009).

[13] Jim Gemmel, Gordon Bell, Roger Lueder, Steven Drucker and Curtis Wong, My Life Bits: Fullfilling the Memex Vision, ACM Multimedia System Journal, pp.235-238(2002).

[14] Abigail J. Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, Ken Wood, Do life-logging technologies support memory for the past?: an experimental study using sensecam, CHI '07, pp.81-90(2007).

[15] 相澤清晴 ライフログの実践的活用:食事ログからの展望, 情報処理 Vol.50 No.7, pp.592-597(2009).

[16] 徳永 清輝, まつ本 真佑, 中村 匡秀, レシート蓄積による消費者向けライフログサービスの考察, 電子情報通信学会技術研究報告, Vol.110, No.281,pp.95-100(2010).

[17] 阿部孝司, 年代光宏, 映像内のテロップ 領域解析によるスポーツ中継映像の場面分 割, 画像電子学会誌,vol.34,no.5,pp.578- 587(2005).

[18] Matthias Kranz, Wolfgang Spiessl, Albrecht Schmidt, Designing Ubiquitous Computing Systems for Sports Equipment, Fifth Annual IEEE International Conference on 2007, pp.79-86(2007).

[19] Gunnar Stevens, Volker Wulf, Markus Rohde and Andreas Zimmermann, Ubiquitous Fitness Support Starts in Everyday's Context, The Engineering of Sport 6, Volume 3. Sheffield: Springer, 2006, pp.191-196(2006).

[20] Michael Beetz, Bernhard Kirchlechner, and Martin Lames, Computerized Real-Time Analysis of Football Games, IEEE Pervasive Computing 2005, Vol.4, No.3 ,pp.33-39(2005).

[21] NIKE+:http://nikeplus.nike.com/plus/.

[22] 原伊吹, 神場知成, 田中二郎, パーソナルクラウドを用いたライフログサービス–テニスレッスン支援アプリケーション"テニスログ", マルチメディア、分散、協調とモバイル(DICOMO2011) シンポジウム, pp.383-391(2011) .

[23] Ig-Jae Kim, Sang Chul Ahn, Hyoung-Gon Kim, Personalized Life Log Media System in Ubiquitous Environment, ICUCT, pp.20-29(2006).

[24] Koh Sueda, Henry Been-Lirn Duh, Jun Rekimot, Social life logging: can we describe our own personal experience by using collective intelligence?, APCHI'12, pp.41-50(2012).

[25] 3 軸加速度センサ WAA010：http://www.atr-p.com/sensor10.html.

[26] Nexus S：https://www.google.com/nexus/ .

[27] BIGLOBE クラウドホスティング：http://business.biglobe.ne.jp/hosting/cloud/.

[28] テニスルールブック 2012, 日本テニス協会.

[29] Google play：http://play.google.com/intl/ja/about/index.html.

[30] アンドロイドプログラミングバイブル：http://www.socym.co.jp/book/757.

[31] あんず文字：http://www8.plala.or.jp/p_doice.

[32] Highcharts：http://www.highcharts.com/.

[33] 中野祐介, 河上 寛, 垂水浩幸, ライフログを共有する Life Networking System, 情報処理学会インタラクション,pp.205-210(2012).

[34] 中川 真紀, 塚田 浩二, 椎尾 一郎, Smart Makeup System: ライフログを用いた化粧支援システム, 情報処理学会インタラクション, pp.121-128(2012).

[35] 浅野 敏郎, 佐藤 邦彦, 浜野 博行, 原 肇, テニススウィングの解析と定量評価, 精密工学会誌, Vol.73, No.2, pp.281-285(2007).

[36] SlamTracker：http://www.australianopen.com/en_AU/ibmrealtime/index.html.

[37] 北原駿, 内田理, 画像処理を用いたテニス競技におけるデータ自動記録・分析手法の提案, 情報処理学会第 74 回全国大会, pp.421-422(2012).

[38] 植村 暢介, 川添 和泉, 仙石 謙二郎, 濱川 礼, テニスにおけるラリーの簡易図化及びプレーシーンの視聴補助システム, 情報処理学会第 74 回全国大会, pp.417-418(2012).