

SensorPipe : スマートフォン側のプログラミングを一切行わないスマートフォン連携プログラミング

鳥山 らいか*

宮下 芳明†

概要. スマートフォンには様々なセンサが搭載されている。センサ情報を用いたスマートフォンと PC との連携を実現するには双方でのアプリケーション開発が必要となる。本稿ではスマートフォンの各種センサ情報を PC に送信し続け、PC 側からもスマートフォンにバイブレーションやウェブサイト表示などの所定の動作をさせる事ができるアプリケーション、SensorPipe を提案する。プログラマはスマートフォン側のプログラミングを一切行わずに、PC でのセンサ情報を用いたプログラミングを行うことができる。また一度起動するとバックグラウンドに常駐するため、他のアプリケーション使用時や画面オフ時でも情報の送受信が可能である。これにより、明るい場所に出てスマートフォンを強く振らないとアラームとバイブレーションが解除できない目覚まし時計や、スマートフォンを振ることでその場にふさわしい様々な音を奏でる楽器などのアプリケーションの開発が容易になる。

1 はじめに

一般的にスマートフォンと PC が連携するアプリケーションの開発には、センサ情報を送信するスマートフォン側と PC 側のプログラムを個別に組む必要があるため、製作時間がかさんでしまう。また専用のツールなどを使うことも多く、ある程度の知識を要するため、未経験者にはハードルが高い。

本稿ではスマートフォンのセンサ情報を PC 側に送信し、PC 側からもスマートフォンに所定の動作をさせることができるスマートフォン用アプリケーション、SensorPipe を提案する。これにより、スマートフォン側のアプリケーション開発を一切行うことなく、センサ情報を利用したプログラミングを PC で行うことができる。さらにバックグラウンドに常駐することにより、他のアプリケーション使用時や画面オフ時にも PC 側から送られる所定動作の命令を受信することができる。

SensorPipe によって、目的の駅についたら PC とスマートフォンの両方から通知されるシステム (図 1) や、振り方やその場の状況に応じて音が変わる楽器アプリケーションなどが構築できる。これらはスマートフォンの GPS や電子コンパスを利用する。

スマートフォンから PC に値を送信するアプリケーションとして TouchOSC[1]が挙げられる。これは

画面上でスライダやボタンを操作することで、スマートフォンを OSC・MIDI コントローラ化するものである。他にも、センサ情報を PC に送信する例としては Physical Sensors for Scratch[2]が挙げられる。これはアプリケーションから送られたセンサ情報を Scratch[3]で利用できるというものである。しかし、これらいずれの場合も、PC 側からスマートフォンにアクションを起こさせる事はできない。

また Firmata[4]では、Arduino[5]に一度書き込みを行ったあとは、PC 側でプログラミングするだけで Arduino と PC 側との連携が可能になる。本システムとの違いは、本システムでは Arduino より身近にあるスマートフォンで実現した点と、一切の電子工作の知識を必要としない点である。

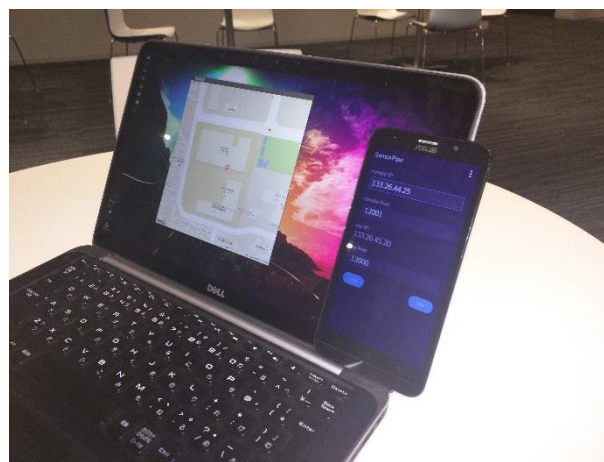


図 1. GPS と電子コンパスを利用した目的地到着の通知アプリケーション

Copyright is held by the author(s).

* 明治大学総合数理学部先端メディアサイエンス学科

† 明治大学総合数理学部

2 システム

システムの使用方法としては、まず同一の Wi-Fi にスマートフォンと PC を接続する。そして、それぞれ自動的に表示される自端末の IP アドレスをもう一方の端末に入力し、任意のポート番号を指定する。最後にスタートボタンを押すと、バックグラウンドでプログラムが実行される。実行されたプログラムはセンサの値の変化やタッチイベント等を検知し、非同期処理によって値を送信する。

SensorPipe はシステムに常駐するため、他のアプリケーションを操作時や画面オフ時であっても送受信が行われ続ける。この時、プログラムがバックグラウンドで起動していることを確認するために、半透明の青い層を画面に表示している。また、もし何かのエラーでプログラムが終了した場合は自動的にバックグラウンドで再起動される。

今回は Processing で扱いやすくするためのクラスをあらかじめ作成した。またスマートフォンと PC との連携には、傾きやタッチの座標などによるリアルタイムな入力を用いられることも少なくない。そのため、通信プロトコルとしてデータ欠損はあるものの、通信速度の速い UDP を採用している。

SensorPipe では、照度・近接・加速度・ジャイロ・方位・磁気などのセンサ情報を扱うことができる。加えて、緯度・経度などの位置情報やタッチの座標などにも対応している。PC 側からスマートフォンに実行させることのできる命令は、トーストやバイブレーション、フラッシュ点灯などの他に、ウェブサイト表示・マップ表示・電話発信などが挙げられる。トーストなどの画面に表示させる命令は画面オフ時には作動しないが、バイブレーションなどは画面とは無関係のため、画面がオフの状態でも作動する。

様々なセンサ情報を送信し続けるため、スマートフォン側にはかなりの処理負担がかかる。そこで SensorPipe では送信回数を削減するため、工夫を行っている。まず、センサ値の変更検出が高頻度なものをまとめて送信することである。そして、連続でタッチイベントの「動く」が発生している場合、3フレームごとに1回タッチ情報を送信することである。

3 ハッカソンの実施

今回、大学生2名を対象に SensorPipe を用いたハッカソンを実施した。各自のスマートフォンへ SensorPipe をインストールし、Processing における送受信データの扱い方についての説明した後に、2時間程度の作業時間を設けた。

作例「ナビたのめっと」は PC 側の地図上でスマートフォン所持者をナビゲーションするプログラムである。PC 側の地図には GPS や電子コンパスの値

に応じたスマートフォンの位置と向きがマーカで示される。PC 側の地図上で目的地をクリックするとスマートフォン側に座標と方向が表示され、緯度・経度が目的地付近になると振動して到着を知らせてくれる。これにより、建物内部のような地図に記載されていない場所へのナビゲーションを実現することができる。

作例「Shake Alarm」は明るい場所で激しく振らなければ止まらないアラームプログラムである。あらかじめ PC 側でセットした時間になるとスマートフォンが振動する。また、同時に PC 側からもアラームが鳴る。加速度センサと照度センサを組み合わせることで、明るい場所で激しくスマートフォンを振る動作を検出し、アラームが停止する仕組みである。屋外へ出て行き、思いきり振らない限り鳴り続けるため、止まる頃にはすっかり目が覚めている。

ハッカソン実施後に、本システムを使用せずに同様のアプリケーションを実装することができるかを被験者2名に質問したところ、1名は不可能であると答え、もう1名は可能ではあるが実装時間が数倍に増加すると答えた。

4 考察

ハッカソンより、プログラマはスマートフォン側のプログラミングを一切行うことなく、スマートフォンと PC の連携プログラミングを実現することができた。また画面を占有しないため、別のアプリケーションと同時並行で実行するといった使い方も可能である。

現段階ではスマートフォン側で送受信できる情報が少ない。より多様なアプリケーション開発を可能にするため、今後も機能拡充を進めていく。

謝辞

本研究は、JST、CREST の支援を受けた。

参考文献

- [1] hexler TouchOSC (2015/10/19 確認)
<http://hexler.net/software/touchosc/>.
- [2] Physical Sensors for Scratch (2015/10/19 確認)
<http://s4scratch.moyashi-koubou.com/>
- [3] MIT Scratch (2015/10/19 確認)
<https://scratch.mit.edu/>
- [4] Firmata MainPage (2015/10/19 確認)
http://www.firmata.org/wiki/Main_Page
- [5] Arduino (2015/10/19 確認)
<https://www.arduino.cc/>