

スマートフォンカーナビ向け加速度センサデバイス

池上翔太^{†1} 石崎新^{†1} 山辺教智^{†1} 北上慎二^{†2}
梶並知記^{†1} 清原良三^{†1}

近年カーナビゲーションシステムは様々なタイプの製品が出荷されており、ほとんどの車に搭載されているといっても過言ではない。しかし、ネットワークを通じて最新の情報を得たり、車両から情報を発信するタイプの通信型カーナビの普及は進まなかった。通信型カーナビは最新の地図を利用し、最新の交通情報が利用できるにも関わらず、そのコストの問題で普及してないと考えられる。しかし、最近ではスマートフォンを利用した通信コスト不要の無料のカーナビの登場、スマートフォンの出力を大画面に表示する”Display Audio”という製品の登場から爆発的な普及が想定されている。しかしながら、ユーザの操作性に関しては従来と変わらず問題点が多い。本論文ではこれらの課題を整理し、加速度センサを利用した解決手法を提案、実装評価する。

HMI for Car Navigation System on a SmartPhone by an Sensor Device

SHOTA IKEGAMI^{†1} ARATA ISHIZAKI^{†1} TAKATOMO YAMABE^{†1}
SHNJI KITAGAMI^{†1} TOMOKI KAJINAMI^{†1} RYOZO KIYOHARA^{†1}

Recently, a lot of types of car navigation systems are shipped for the world-wide markets. However, car navigation systems which have communication functions by cellular network are not widely spread. Though these types of car navigation systems can use newest maps and traffic jam information, these devices are not used very much because the users have to pay the communication costs. However, the free navigation applications are running on the smartphones and “Display Audio” which has large display and connected for the smartphones. Therefore these types of devices will be spread. However, there are some problems for HMI. In this paper, we propose a new user interface for car navigation systems which use an accelerometer device.

1. はじめに

近年、カーナビゲーションシステムはオーディオシステムと一体化されることが増えており、製造時に組み込まれるものや製造後独自購入をするものを合わせるとほとんどの自動車に搭載されているといっても過言ではない。

しかし、これらのカーナビゲーションシステムは地図を最新の物に更新する際などディーラに行くか、DVDを購入して時間をかけて更新しなくてはならず、また多大な追加コストが掛かる場合も少なくない。

また、通信型カーナビでは、テレマティクスサービスと言われる有用な情報を受信するサービスが利用でき、地図も常に最新状態に保てるなど普及が期待されたが、通信コスト負担の問題もあって、それほど普及していない。

その一方で、スマートフォンは普及が進み、ある程度正確な位置特定ができ、最新の地図と渋滞情報が利用できる

環境が整ってきた。追加コストのかからない無料のカーナビゲーションアプリも登場しているが、運転中に見るには画面が小さく、操作も信号停止中などの短時間では慣れない人にとっては難しいという問題もある。

しかし、図1に示すように、スマートフォンを Display



図1 Display Audio の例
「TOYOTA Touch Life」

†1 神奈川工科大学
Kanagawa Institute of Technology

†2 三菱電機 (株)
Mitsubishi Electric Corporation



図2 従来のカーナビゲーション操作風景

Audio と呼ぶ画面とオーディオに特化したデバイスに USB や Bluetooth による無線接続し、画面を大きく表示させる製品が登場しつつある。これらは、Mirror Link[1] や MiraCast[2] と呼ばれるプロトコルを利用して、デバイス側とスマートフォン側でアプリが同期をとって動作し、大きな画面上であたかもスマートフォンを操作しているかのように見せることができる。

このようなコストの安いデバイスの登場により、大衆車市場においては、スマートフォンを用いたカーナビというソリューションが普及することが容易に想像できる。しかしながら、スマートフォン上のアプリケーションと Display Audio 上のアプリケーションが同期をとって進化するとは考えにくく、自動車の寿命を想定したカーナビの購入においては、安価な Display Audio とはいえ、課題があると考えられる。

一方、Google 社などはスマートフォンの画面のみを大画面に表示し、操作も大画面上で行える技術[3]を開発しており、このような解が普及することも考えられる。しかし、図2に示すような従来同様のカーナビの操作をさせていたのでは、ドライバーの注意散漫が起こる可能性があり、安全性に疑問が生じる。

そこで、本論文では、このようにスマートフォンの画面がそのまま大画面に表示され、その大画面上で操作するようなカーナビゲーション装置を前提として、注意散漫になりにくい操作インターフェースを提案し、実装評価する。

提案手法では、手の動きだけでタップやダブルタップ、フリックといった操作を行い、加速度センサを利用してこの判別を行う。

2. スマートフォンカーナビの UI 要求

カーナビは、自動車がたとえ停止中を想定するとしても、運転中に操作することを視野に入れておく必要がある。そのため大まかに分けると以下の3つの要求を満たしている必要がある。

- 1). 安全性
- 2). 快適性
- 3). 正確性

これらそれぞれに関して要求を整理する。

2.1 安全性

安全性を考慮するために、法律面や、業界の自主規制[4]などがあり、停止中（ギアをニュートラルやパーキングに入れ、サイドブレーキを引いた状態とする）、一時停止中（速度が出てない状態で、信号停止中など）、走行中などの状態に応じて、操作の可、不可が決まっている。例えば、TV の視聴など動画の視聴は停止中でなければならぬ。また、音楽の選択などは一時停止中か停止中でなければならぬといった制限がある。

Display Audio が車載ネットワークに接続されると考えると、ギアの位置情報や速度の情報、サイドブレーキの情報は取得できるため、それぞれの状態に応じた UI を作成すること十分可能である。そのため、安全性を考慮した UI への要求は以下に示す内容であると考えられる。

- 1). 各種状態に応じて、操作を許可、不許可ができること。
- 2). 運転中の操作は目や体が無理にならないように、最小限の機能の操作できること。
- 3). 一時停止中も、すぐに操作が完了するような UI とすること。

2.2 快適性

快適性は、安全性と表裏一体かもしれない。所望の情報を見るための操作が最少限の操作でできることが望ましい。また所望の情報を見ることができるとも快適性の一つであると考えられる。以下に、快適性のための要求を示す。

- 1). 運転車のまわりに余計なケーブルなどを引かないこと。
- 2). 所望の情報を見るまでの操作ができる限り少なくなること。
- 3). 無理な姿勢などを取る必要がないこと。
- 4). できるだけ普段利用するスマートフォンと同じ操作で所望の情報が見れること。

2.3 正確性

正確性とは、正確に操作できることを示しており、例えばタッチパネルなどですぐに間違ったところを押下と判断してしまうような UI は正確性に欠けると考える。あるいは、ダブルクリックをしたつもりがシングルクリックになるなど意図した操作と認識されないようなことがあってはならない。即ち、似たような動作をとったときに誤認識を起こさないよう明確に操作が区別できることが要求である。

2.4 要求に対する実現方式

本論文では上記に述べた UI の要求を満たす方式として以下の特徴を持つ UI を提案する。

- 1). 運転者が姿勢を変えず、視線も注視が必要のないように、ワイヤレスの指輪型の加速センサ+角速度センサを利用したジェスチャー型の UI を提案する。ジェスチャーはスマートフォンの UI とほぼ同様の指の動作を認識させる。
- 2). 快適性の要求を満足するために、スマートフォン上では独立のアプリとなることが多い、ナビゲーションアプリ、音楽アプリ、ブラウジングなどを区別せずにすむように、HTML5 と CSS3 を利用したアプリは独立しているにも関わらず、一体のアプリとして見える方式とし、操作数が少なくなるようにする。
- 3). コンテキストウェア技術を利用し、ドライバの状態（いつも運転している場所か、あまりいかない場所か、はじめての場所か）、交通状況（渋滞情報、路面情報）、運転車の嗜好に応じた情報をデフォルトで表示することにより操作数が少なくなるようにする。

既に HTML5, CSS3 での実現に関しては文献[5]、[6]でも報告しており、本論文では、ジェスチャー型での認識方式に関して提案、実装、評価を行う。



図3 Leap Motion (Leap Motion 社)



図4 Leap Motion での実験

3. 関連研究

3.1 遠隔操作

離れた場所に設置された端末に対して触れることなく操作を行う方法として以下に示すような手法がすでに提案され、製品化されている。それぞれの特徴と課題に関して述べる。

1). 可視光カメラによる認識

自動車には、ミラーなどに可視光のカメラを設置していることが多い。しかしながら、目線を追いかけるなどが主な用途である。このカメラを意識してジェスチャーをする手法が考えられるが、夜など可視光では利用できないケースが課題となる。

2). 赤外線カメラによる認識

複数の赤外線カメラを用いてジェスチャーを認識することのできる図3のような製品(Leap Motion)[7]がある。夜中でも使えるため、有力な候補である。図4に示すように、試行的に利用してみた結果、Leap Motion は認識精度が高いが、赤外線センサであるため指同士が重なったり、センサから角度がずれたところでジェスチャーをすると正常に認識出来ず、車上で且つ手元で操作出来る範囲で赤外線認識しなければならないため状況を考える利用に無理があり、快適性を失うと判断された。

3). 音声認識

方言や個々の声質によって認識精度が変わる上、オーディオによる音や会話によって認識が阻害される可能性があるが、最近のスマートフォンの音声認識技術の進歩を考えると、音声認識とジェスチャーの両方を備えることが望ましいと考える。

4). 加速度センサによる認識

加速度センサを利用した UI についても種々の研究がある。文献[8]では、ウェアラブルコンピューティングにおけるポインティングデバイスとして2個の加速度センサを利用したポインティングデバイスが提案されている。あくまでポインティングデバイスであり、そのままタップやダブルタップの操作には応用できない。また、文献[9]では指輪型に近いデバイスを提案しているが、指そのものの動作を再現するには使いにくいことが示されている。

3.2 操作性

また、モバイルデバイスのユーザインタフェースとしての操作方法に関する研究も多数ある。例えば文献[10]では情報の移動を直感的に表現するインタフェースが提案されている。

本研究では情報の移動ではなく、見える情報の変更に関

する操作に限定されるため、この方式が良いわけではない。このような状況から自動車の中の限定した環境では、加速度センサと角速度センサというようなコストの安いデバイスを利用して少数の操作を実現することは十分可能であると判断するとともに、従来の提案されている方式だけではそのまま適用できないと判断し、本論文では指輪型のデバイスを意識した限定的操作手法に関して報告する。

4. 操作判定方式

6軸センサ(ZMP社製IMUZ2)を図5に示すように手の甲に取り付け、各種操作に対応する動作を行い加速度と角速度の各軸における波形データを取得した。また本実験ではタップ、ダブルタップ、上下左右のフリックの3種類の動作を行った。本来は指に装着する指輪型センサ[10]などを利用すべきであるが、試作としては手軽に利用できるため、手の甲につけるタイプで実験を実施した。

本実験ではタップ、ダブルタップ、上下左右のフリックの3種類の動作を行った。これらの波形のピーク値や谷の数値の差から各動作の判別を行えるか確認した。



図5 加速センサデバイス

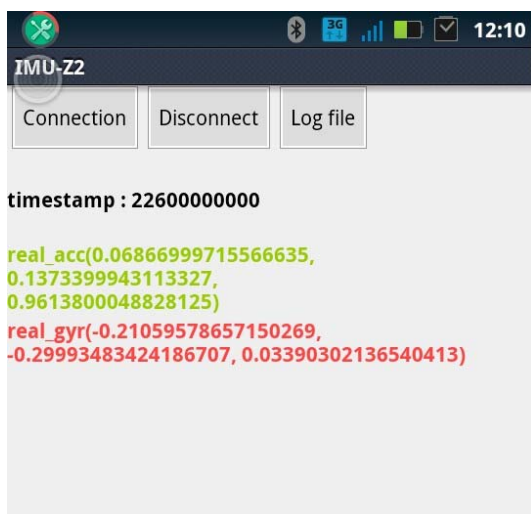


図6 加速度と角速度の値測定

また後述の判定には加速度、角速度を android 上のアプリケーションを利用し、数値測定を予備実験を行うことにより定めた値を使用している。

4.1 タップ動作の判定

加速度、角速度の xyz 軸の値から判断基準となる値を決め、その値を超えたときにタップであると判定する。以下のフローに沿って判定する。

- 1). 加速度の xyz 軸の値がそれぞれ
 X:-0.400 以上、Y:-1.000 以上、Z:0.800 以上の値
- 2). 1)の後 400ms 以内に加速度の値がそれぞれ
 X:-0.350 以上、Y:0.150 以下、Z:0.750 以下の値

また、後述のダブルタップの操作がないことを待って、タップ判定とする。これらの閾値は、予備実験を行い、図6に示すようなデータを取得した上で、適切な値として決定し、利用している。

4.2 ダブルタップの判定

前述のタップ動作が行われてから一定時間内に再度タップ動作が行われたときにダブルタップであると判定する。以下のフローに沿って判定する。

- 1). 加速度の xyz 軸の値がそれぞれ
 X:-0.400 以上、Y:-1.000 以上、Z:0.800 以上の値
- 2). 1.の後 400ms 以内に加速度の値がそれぞれ
 X:-0.350 以上、Y:0.150 以下、Z:0.750 以下の値

-----ここまでがタップ判定-----

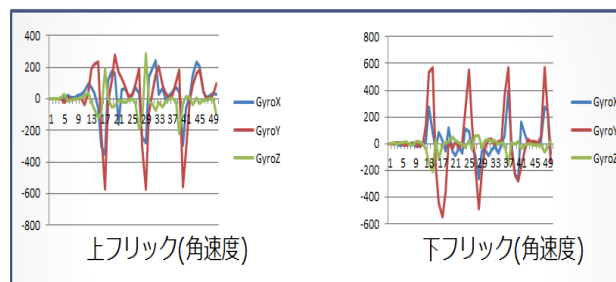


図7 角速度による上下フリックの判定

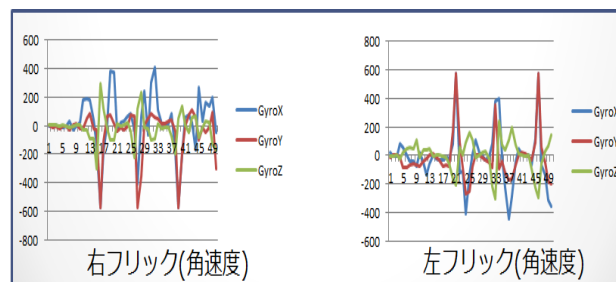


図8 角速度による左右フリックの判定

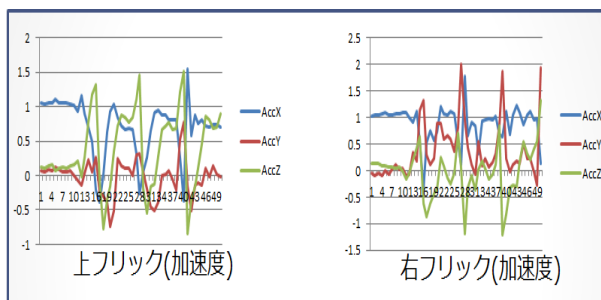


図9 加速度による上フリックと右フリックの判別

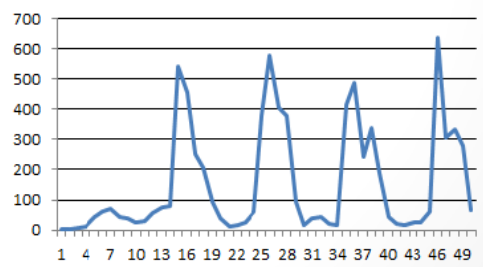


図11 角速度の絶対値によるタップの判定

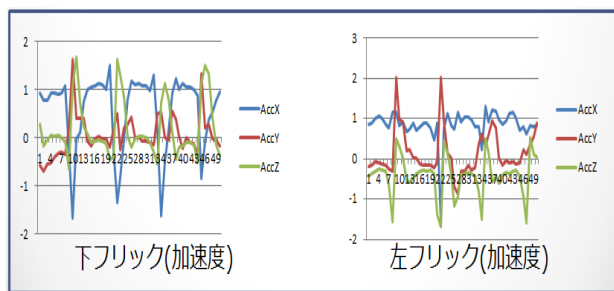


図10 加速度による下フリックと左フリックの判定

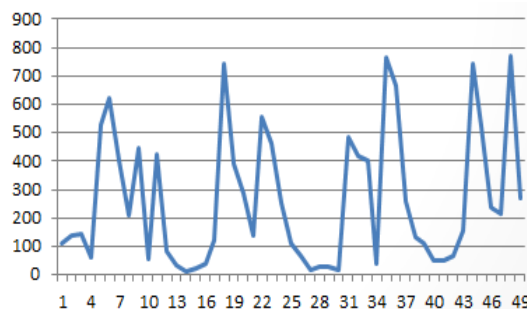


図12 角速度によるダブルタップの判定

3). 1),2 の後 600ms 以内に再度同じ判定が起こる

加速度、角速度の波形を比較して上下左右のフリックを判別する。

本判定においても待ち時間などは何度も予備的な実験を行った上でその値を決定している。

4.3 上下左右フリックの判定

図 7,8 に示すように、角速のみでは、右フリックと上フリックの区別がつかない。また左フリックと下フリックの区別がつかない。そこで、角速度にさらに加速度センサの値を加味する以下のフローでの判定を行った。

- 1). 角速度の値から上下フリックと左右フリックを判別する。(図 7, 図 8)
- 2). 加速度の値から上フリックと右フリック、下フリックと左フリックを判別する, (図 9, 図 10)

4.4 センサの装着向きによる違いへの対策

前項で用いた加速度や角速度では、デバイスの向きが違う場合など使えない。そこで、値の絶対値の合計を用いて判別を行うべきである。

4.4.1 タップの判定

角速度の絶対値の合計から判断基準となる値を決め、その値を超えたときにタップであると判定する。今回は以下のフローに沿って判定を行う (図 11)。

- 1). 角速度の絶対値の合計が 500 以上の値
- 2). 1 の後 500ms 以内に絶対値の合計が 100 以下の値

4.4.2 ダブルタップの判定

前述のタップ動作が行われてから一定時間内に再度タップ動作が行われたときにダブルタップであると判定する。今回は以下のフローに沿って判定を行う (図 12)。

- 1). 角速度の絶対値の合計が 500 以上の値
- 2). 1 の後 500ms 以内に絶対値の合計が 100 以下の値
-----ここまでがタップ判定-----
- 3). 1,2 の後 600ms 以内に再度同じ判定が起こる

4.5 実験結果

判定基準となる値を決め、前項の手法を用いて判定実験を行った結果判定は可能であることがわかった。しかし外部のセンサから Bluetooth 経由で送られてきたデータをスマートフォンで読み取り処理をしているためタイムラグが発生するが、画面の切り換え程度であり、そうストレスにはならないと考える。

5. 考察

現在配信されているカーナビゲーションアプリはスマートフォンのタッチ操作によって動作させるものしかない。そのため外部センサから送られてくるイベントを反映させ



図 13 カーナビゲーションアプリのオーディオと地図の画面

ることのできる機能を持った新しいカーナビゲーションアプリを用意しなければならない。そこで我々は HTML5、JavaScript、CSS3 を用いたカーナビゲーションアプリと本機能とを合わせるべきであると考え。

また、これにコンテキストウェア技術を組み合わせることにより操作そのものが不要になるケースが増加すると考える。

5.1 カーナビゲーションアプリ概要

前述したように HTML5 をベースとし JavaScript、CSS3 を組み合わせて Google MAP を用いたナビゲーション、オーディオ、SNS ブラウザの 3 つの機能を持たせたカーナビゲーションアプリで操作数を少なくし直感的に動作させることを目的に作成した。

従来のカーナビゲーションは地図を表示している状態からオーディオの曲を選択しようとする画面を切り替えて行う必要があるが、本カーナビゲーションアプリは HTML5 のオーディオ機能と JavaScript のプレイリスト機能を用いるため他のオーディオアプリを使用せずとも十分な操作が行え、また地図とオーディオを両立させているため操作数が少なく済む。

画面の切り替えも CSS3 のタブ機能を用いているためタップのみで瞬時に切り替えることが可能となっている。

5.2 カーナビゲーションアプリの評価

それぞれの機能を持った画面をタブによる操作で切り替

えていくため操作数は従来のカーナビゲーションやスマートフォンのアプリケーションに比べて操作数は減っている。

またコンテキスト情報に応じて表示する画面を切り替えるという機能を持たせることでニーズに合った場合の操作数はより少なくなるが、合わなかった場合は操作数が増える結果となるため実際に試してもらうことにより評価する必要がある。

6. 今後の課題

現在以下の事柄を目下の課題として挙げている。

- 1). タップやフリックといった単純な動作でも操作時の手の動きは個人差がでてくるため複数人の操作のデータを取得し、誰が動かしても正しく認識するように動きの範囲を決めていく
- 2). 手の動きを用いた操作は項目の移動や決定といった基本動作は手軽に行うことができるが、地図の検索などを行う際の文字の入力には難があるためどういった手法で補完するかを検討する。
- 3). センサからのデータの取得を始めるタイミングをどのようにして決めるのか(アプリが起動した時点からなのか、アプリ内にボタンを設置し押された段階からなのか)を検討する。
- 4). 前項で提案したカーナビゲーションアプリと組み合わせの際にセンサから送られてくるイベントをカーナビゲーションアプリ側とどのような手法で参照しあうのかを検討する。
- 5). 4.5 項で述べたようにスマートフォンの読み取り時に発生するタイムラグに加えてカーナビゲーションアプリにデータを送る際のタイムラグも加算されると考えられるため、操作に違和感がある程度のもだった場合改善策を講じる必要がある。

7. おわりに

本論文では、現状のカーナビを整理し、通信型カーナビゲーションシステムからスマートフォンベースのカーナビに進化していくことを示した。その場合の操作性の課題に注目し、直感的な操作、操作数の削減、ユーザの所望する情報の自動提示の手法を示し、その中でも直感的な操作に関して、加速度と角速度の情報を利用した操作方式を提案した。

コストの安い装置を手の甲に装着することにより、直感的な操作としてのタップ、ダブルタップ、フリックの判定が十分にできることを示すとともに、自動車の停止時や運転時の数少ない操作にはそれで十分対応できることを示した。

今後コンテキストウェア機能の実装とともに、操作画面上の課題などを整理してさらに進化させる予定である。またユーザに対してアンケート調査などを行い、コンテキ

ストアウェア技術でなくとも、デフォルトでどのような画面が良いのか検討をすすめ、市場にて評価されるようなスマートフォンベースのカーナビゲーションシステムを開発する予定である。

参考文献

- 1) CARCONNECTIVITY: <http://www.mirrorlink.com/>
- 2) Wi-Fi CERTIFIED Miracast:
<http://www.wi-fi.org/wi-fi-certified-miracast%E2%84%A2>
- 3) Matei Negulescu, Yang Li, "Open Project: A Lightweight Framework for Remote Sharing of Mobile Applications, ACM Symposium on User Interface Software and Technology (UIST 2013), 2013
- 4) 日本自動車工業会 : <http://www.jama.or.jp/>
- 5) Takatomo Yamabe, Shota Ikegami, Arata Ishizaki, Shinji Kitagami, Ryoza Kiyohara: Car Navigation User Interface Based on a Smartphone, The seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU 2014)
- 6) 山辺 教智, 石崎 新, 池上 翔太, 梶並 知記, 清原 良三, " スマートフォン向けカーナビのインタフェース," マルチメディア通信と分散処理ワークショップ (DPSWS2013)
- 7) Leap Motion: <https://www.leapmotion.com/>
- 8) 大坪由香利, 中井 満, "加速度センサの傾き情報を併用した空中手書き文字認識," 電子情報通信学会 PRMU MVE IPSJ-CVIM pp.263-268 (2013)
- 9) 所洋平, 寺田努, 塚本晶彦 "2つの加速度センサを用いたポインティング手法" WISS 2008 (2008)
- 10) 峰健三, 大淵竜太郎, "携帯情報機器のための入力デバイス DigiTrack," WISS 2001(2001)
- 11) Ring: <http://logbar.jp/ring/>