

スマホカーナビにおける SVM を利用した車の停止判定方法

浅見 宗広[†] 大平 雄貴[†]

ヤフー株式会社 メディアカンパニー生活メディア事業本部[†]

1 はじめに

スマートフォンの普及に伴い、車載カーナビと同等な機能を備えたカーナビアプリも普及している。しかし、GPS が届かないトンネル内では現在位置推定が困難なことが大きな欠点である。そこで、スマートフォンに搭載されているセンサーを利用して車速を推定し課題解決する方法が考えられるが、それには、車が走行中か停止中かを判定することが重要となる。本研究では、Support Vector Machine(以下、SVM)を利用した停止判定方法を提案する。

2 従来方法

従来の車の停止判定方法は、加速度センサーから定期的(20ms など)に3軸加速度値を計測し、1秒間での分散が、ある閾値以下であれば停止と判定する方法である[1]。ただし、最適な閾値をどう設定するか課題が残る。

3 提案方法

スマホカーナビでは、停止判定に影響を与える環境要因として主に以下が考えられる。

- A) 車の振動
- B) スマホの車への設置姿勢
- C) クレードルなどの設置時のがたつき

これらの環境要因を吸収するには個別に対応するのが最適と考え、SVM により継続的にモデルを更新し、停止判定する方法を採用した。

3.1 特徴量

3軸加速度の計測値は、スマホの車への設置姿勢に依存する。その影響を受けにくいように1秒間で次のように2成分の加速度成分を考える。

- A) 3軸加速度の各成分を平均した方向の成分
- B) 上記と垂直な成分の大きさ

SVM の特徴量は、上記の各2成分について 1) 平均値、2) 標準偏差、3) 最大値、4) 最小値、をとり合計で8個設定する。

3.2 学習データ

車が停止状態かどうかは、基本的には GPS センサーから取得できる速度データ(以下、GPS 速度)を利用する。つまり、トンネルに入る前には必ずトンネル外で GPS センサーを捕捉した状態で、ある程度の走行が必要となる。GPS 速度が1km/h 以下であれば停止状態、3km/h を超えれば走行状態として学習データを収集する。停止、走行それぞれの学習データが60個、120個、180個集まった段階でモデルを構築し、180個集まった段階で一旦学習データを破棄し(リセット)、再び収集、モデル構築を繰り返す。これにより最新の環境に対応したモデルに維持されると期待できる。なお、学習データがリセットされるまでは最新のデータが優先され保存される。

なお、これらの学習処理は OSS である LIBSVM[2]を採用し、Linear カーネルを用いた。

3.3 最適モデル選択

学習後構築したモデルは8個保持され、速度帯毎に収集された表1のテストデータにそれぞれ適用され、判定の正解率が最も高いモデルをその時点の判定モデルとして採用する。

表1: テストデータの仕様

帯 ID	0	1	2	...	10
速度範囲	0~5	5~15	15~25	...	95~
サイズ	256	24	24	...	24
状態	停止	走行			

4 評価方法

カーシェアリングサービス等により車を手配し、東京、名古屋、岐阜で実際に車を走行させ独自に開発したアプリにより加速度、GPS 速度等をログに保存した。また、GPS 速度では評価精度に問題があるため、速度の正解データとして OBD2 を利用し専用アプリ[3]からログ保存した。スマホは、特に設置姿勢を意識せず見やすいようにクレードル等により車に固定した。

これらのログに対して、従来方法と提案方法による処理を実行し比較検証した。OBD2 から取得した速度を正しい速度とし、今回は2km/h 未満を停止状態、2km/h 以上を走行状態と定義した。

加速度等のログと OBD2 のログは別端末を利用したため、同じ UNIX time によるタイムスタンプでもずれが発生する。このずれは OBD2 速度と

Vehicle Stopping Determination Method Using SVM for Smartphone Car Navigation
Munehiro Azami[†], Yuki Ohira[†]
[†]Daily-Life Media Division, Media Company, Yahoo Japan Corporation.

GPS 速度に対して最小 2 乗法を適用して補正した。従来方法では、3 軸の加速度成分に対して 1 秒毎に分散を求め、3 つの分散値が全て 0.001 未満の場合、停止と判定、それ以外は走行と判定した。ただし、加速度は重力加速度を単位に計測した。

停止判定正解率 R_s 、走行判定正解率 R_m 、および総合判定正解率 R_t を次のように定義する。

$$R_s = N_{cs}/N_s * 100$$

$$R_m = N_{cm}/N_m * 100$$

$$R_t = (N_{cs} + N_{cm}) / (N_s + N_m) * 100$$

ただし、各パラメータの意味は表 2 の通り。表 3、表 4 には実走行条件と LogID の定義、スマホ端末と DeviceID の定義をそれぞれ示した。

表 2 : 各パラメータ

パラメータ	説明
Ncs	停止判定正解数
Ncm	走行判定正解数
Ns	停止状態総数
Nm	走行状態総数
Nt	全状態数 (Ns + Nm)

表 3 : 実走行の条件と LogID 定義

*a	場所	*b	車
T1	名古屋	95	HONDA FIT
T2	東京	84	TOYOTA COROLLA FIELDER
T3	岐阜	105	TOYOTA AQUA
T4	東京	107	Mazda DEMIO

*a) LogID, *b) 概算走行距離 [km]

表 4 : スマホ端末と DeviceID 定義

DeviceID	端末	DeviceID	端末
4K	404KC	P5	iPhone5
N5	Nexus5	P6	iPhone6
K3	KYV35	6p	iPhone6 plus

5 結果とまとめ

表 5、図 1 に各正解率を示した。Rs については、SVM 利用時が優位な傾向にあった。Rm、Rt については従来手法が優位な場合もあった。

注目すべきは、LogID が T2、DeviceID が N5 の結果である。従来手法での Rs が極端に低いケースにおいても SVM 利用の提案手法では 94% と高い正解率となった。これは従来方法では対応できない特殊な環境下でも SVM の学習処理によって高い判定精度を維持できたと推測できる。

ユーザーがどんな環境下でアプリを利用するか予測がつかないため、提案手法が信頼性の観

点では優位であると考えられる。

表 5 : 各正解率の結果

*1	*2	Rm		Rs		Rt	
		分散	SVM	分散	SVM	分散	SVM
T1	P6	99.7	98.7	80.9	84.8	98.6	97.9
T2	4K	64.6	69.1	96.4	97.6	73.1	76.8
	N5	98.4	85.3	42.4	94.0	81.7	87.9
T3	K3	97.9	97.6	96.1	97.6	97.6	97.6
	N5	97.7	93.8	87.7	95.1	95.4	94.1
	N5	99.6	99.6	94.0	96.9	98.5	99.1
T4	P5	93.7	90.3	89.6	95.4	92.1	92.2
	P6	89.2	93.1	97.2	96.9	92.2	94.5
	6p	85.4	92.7	96.4	97.1	89.6	94.4
	N5	90.2	89.3	96.7	96.8	92.6	92.1

*1) : LogID, *2) : DeviceID

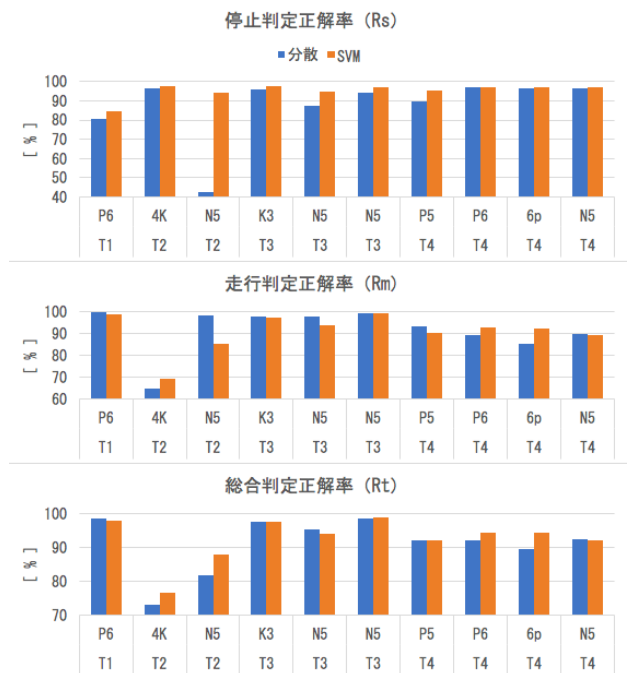


図 1 : 各正解率の結果

参考文献

- [1] 木山、高橋、祖父江、相川 : 「傾斜したスマートフォンによる自動車の 3 軸加速度算出手法」、情報処理学会マルチメディア、分散協調とモバイルシンポジウム 2014 論文集, pp. 16-23.
- [2] LIBSVM: <https://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>
- [3] OBD Info-san! MAXWIN: <http://www.maxwin.jp/content/obd/m-obd-v01.html>