

車載カメラの映像と前面展望映像の重ね合わせによる 鉄道模型の新たな楽しみ方の提案

田中 裕人¹ 塚本 昌彦¹ 寺田 努^{1,2}

概要：鉄道には様々な楽しみ方があるが、新たな楽しみ方として鉄道模型の車載カメラの映像に実車の前面展望映像を重ね合わせることを提案する。このシステムではセンサを搭載した模型車両を前面展望映像の重ね合わせを行いたい部分を1度走行させることによって、加速度と角速度から線路形状を認識する。重ね合わせる前面展望映像はあらかじめエッジ抽出を行い、絞り込みを行うことによって線路形状を推定しておく。映像の中から模型の線路形状に適したものを検索する。2回目以降は車載カメラの映像に線路形状の適した前面展望映像を重ね合わせることによって運転士気分を味わえるようにする。

A New Way of Enjoying Railroad Models by Mixing Real-time Camera Image Taken on Model Railroad Train and Stored Front View Image Taken on Real Train

TANAKA HIROTO¹ TSUKAMOTO MASAHIKO¹ TERADA TSUTOMU^{1,2}

Abstract: There are many kinds of train fans who enjoy them in various ways. In this paper, we propose a new way where a system mixes real-time camera image which is taken on railroad model and front view image which is taken on real train and is stored on a computer. A system configuration is done by once making a railroad model train run 1 round attaching with an accelerometer, and the system analyzes the sensor data to estimate the track form. Stored front view images taken on a real train are also analyzed to detect the track form in front of the train beforehand. When a user makes the model run again, the system selects an image from stored images whose track form is similar to the current situation of the model and mixes the image with the real-time camera image taken on the model to show the user richer images as if he/she becomes a driver.

1. はじめに

現実の電車へのIT(Information Technology)の導入が行われているにも関わらず、鉄道模型は発光ダイオード(LED: Light Emitting Diode)の電飾が導入されるに留まっており旧態以前のものである。一方でカメラを搭載した模型が発売されたり[1]、センサやデコーダといわれる集積回路(IC: Integrated circuit)チップなどの電子機器と組み合わせることにより、同一の線路上の列車を実物同様に個々に制御したり[2]、自動運転も行われるなど鉄道模型の可能性

はより広がったと考えられる。鉄道ファンは前面展望映像を見たり、模型を作ったり、運転したり、シミュレーションで運転の疑似体験などをすることによって楽しんできた。

そこで本研究では、鉄道模型を情報機器と連携させることで、「電車でGO」のようなリアルな運転体験をジオラマ上を走らせることによって楽しめるシステムを提案する。提案システムでは、鉄道模型(NゲージまたはHOゲージ)と小型ハイブリッドセンサ、コントローラを用いて重ね合わせたい部分を鉄道模型を走行させることによって線路の形状を認識し、似た線路形状の前面展望映像を検索する。似た線路形状の前面展望映像があれば、2回目以降は鉄道模型の速さに合わせて再生する。これにより、大掛かりなシステムも要らず、前面展望映像と模型を用いて運転士目

¹ 神戸大学大学院工学研究科
Graduate School of Engineering, Kobe University
² 科学技術振興機構さきがけ
PRESTO, Japan Science and Technology Agency

線でのリアルな運転を味わえるようになる。ジオラマ製作時にはこのシステムを用いることによって、イメージを膨らますことができる。また、これまであまり交流のなかった分野のファンが同次元で楽しめるようになる。

以降、2 章では関連研究について述べ、3 章ではシステムの要件および構成について詳細に述べる。4 章では本システムの実装を行い、最後に 5 章でまとめるを行う。

2. 関連研究

近年、バスやタクシーにはドライブレコーダを搭載したものが増えており、都電荒川線では車載映像記録装置 [3] がつけられている。また介護用ロボット等が道を把握するために、カメラによって位置や速度の推定を行うものがある。位置を推定するものとして小野らの「車載カメラ映像の時空間マッチングによる自車位置推定」[4] があげられる。これは GPS に頼ることなく、あらかじめ蓄積された車載カメラ映像とマッチングするによって自車位置を推定するものである。条件によっては誤差が 2 (m) も出るので、模型には適していない。また、位置・速度を推定するものとして木口らの「車載カメラ画像による位置・方位・速度の測定」[5] があげられる。これは角を検出することによって位置・方位・速度の推定を行っているため、角の存在しない鉄道では利用できない。

鉄道とカメラを用いたものとして池ヶ谷らの「可変テンプレート画像による車両前方のオプティカルフローの検出」[6] や鶴飼らの「光学画像センサによる列車前方の監視手法」[7] があげられる。前者はテンプレート画像を可変にすることによってオプティカルフローをもとめるもので、後者はテンプレートマッチングによってカメラから得られた画像の線路の形状を把握し、線路が続いているかを判断することによって線路上に異物がないかを判断するものである。

鉄道模型と車載カメラを用いたものとして「列車運転士支援システムの開発」[8] があげられる。これは画像認識用の標識を模型車両の車載カメラによって動画撮影し、リアルタイム処理した後、標識で決められた音を自動的に鳴らすものである。提案するシステムとは用いている車載カメラのシステムが異なる。

DTW(dynamic time warping) を映像に用いたものとしては Cuntoor らの「Combining multiple evidences for gait recognition」[9] があげられる。これはシルエット画像からヒストグラムを抽出して DTW による照合を行っている。

3. システム設計

3.1 提案システムの構成

本研究のシステムの概要を示す。加速度と角速度センサとカメラを積んだ鉄道模型、パソコン、コントローラを用いる。模型の加速度、角速度と車載カメラの映像をパソコ

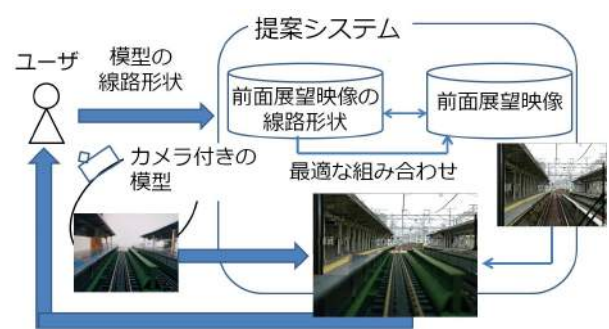


図 1 システムの流れとイメージ

Fig. 1 Flow and image

ンに送る。加速度と角速度から速度と曲線の半径を求め、線路形状を推定する。また、重ね合わせる前面展望映像についてはあらかじめデータ化しておく。線路認識手法を用いて線路形状を推定する。その後オプティカルフローを用いて相対速度を求める。背景が止まっている場合は前面展望映像側の列車の速度を求めたことになる。線路形状の推定結果と合わせて列車の速度の推定を行う。それぞれの線路形状の推定結果を用いて、DTW でマッチングを行う。DTW を用いることで速度をあまり考慮することなく線路形状を比較できる。システム利用者は図 1 のような重ね合わせた映像をパソコンを通して見ることになる。

3.2 システム要件

提案システムの要件を以下に述べる。

模型列車の速さおよび線路形状の推定

線路の形状を把握したり、前面展望映像を模型の速度に合わせて再生するには、模型の列車の速さを把握する必要がある。また、前面展望映像と車載カメラの映像の線路の形状が似ていないと不自然になってしまう。そこで線路の形状を外観を損ねることなく推定することが必要である。

前面展望映像の速さおよび線路形状の推定

前面展望映像の速度および線路形状がわからないと、模型に合わせて再生することができない。よって前面展望映像の速度および線路形状を推定する必要がある。

模型と映像のマッチング

模型と映像の線路形状が似ていないと不自然になってしまう。そこで、より似たものを選択する必要がある。

3.3 模型の予備実験

3.3.1 実験方法

模型の加速度と角速度を計測することによって、速度と曲率が推定できないかを調べるために予備実験を行った。システム構成を図 2 に示す。小型ハイブリッドセンサ waa006 を積んだ鉄道模型 (HO ゲージ) パソコン、コントローラを用いた。小型ハイブリッドセンサの仕様は (表



図 2 予備実験のシステム構成

Fig. 2 System configuration of preliminary experiment

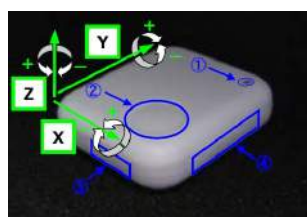


図 3 センサの軸、向き

Fig. 3 Direction

(<http://www.atr-p.com/sensor06.html> より引用)



図 4 センサの搭載方法 (HO)

Fig. 4 Load a model(HO) with sensor

表 1 小型ハイブリッドセンサ waa006 の主な仕様

Table 1 Specification of sensor waa006

CPU	ルネサステクノロジ社 H8S 14.7456MHz
サイズ	39mm(W)×44mm(H)×12mm(D)
重さ	20g
稼動時間	約 6 時間
通信機能	Bluetooth Ver.2.0+EDR (バンド幅: 最大 700kbps, 通信距離: 最長 10m)
加速度センサ	± 2G/4G, 最大 500Hz (日立金属社製)
角速度センサ	X 軸, Y 軸 ± 500deg/s (InvenSense 社製) Z 軸 ± 300deg/s (エプソントヨコム社製)

1) のように、軸および + の向きは図 3 のようになる。模型はコントローラで制御し角長長方形の線路を走行させた。また速さを停止→ゆっくり→はやく→ゆっくり→停止と変化させた。加速度と角速度をサンプリング周波数 100 (Hz) で計測した。小型ハイブリッドセンサは図 4 のように搭載した。

3.3.2 実験結果

図 5 に加速度を、図 6 に角速度を計測した結果を示す。

3.3.3 考察

模型の速さの推定

図 5 の Y 軸の振幅に着目すると速さを大きくすると

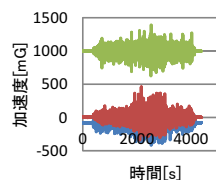


図 5 加速度

Fig. 5 Acceleration

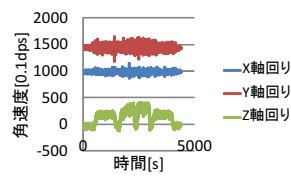


図 6 角速度

Fig. 6 Angular velocity



図 7 前面展望映像

Fig. 7 Foront view image of real train



図 8 グレー化

Fig. 8 Gray image of front view

振幅も大きくなり、速さを小さくすると振幅も小さくなっていることがわかる。このことより模型の速さは Y 軸の加速度の振幅を用いて求めることができる。

模型の曲率の推定

図 6 の Z 軸まわりの角速度に着目すると直線と曲線の区別ができることがわかる。また曲線の場合、加速度から速さの推定ができるので角速度から線路の半径を推定することができる。さらに図 6 の Z 軸まわりの角速度が 0 (dps) を境にして、進行方向むかって右カーブか左カーブかを判別することもできる。

3.4 テンプレートマッチングを用いた前面展望映像の線路の形状推定の予備実験

3.4.1 実装

レール検出アルゴリズム [7] を参考にして、テンプレートマッチングによる線路形状の推定を行った。アプリケーションの開発環境は Microsoft Visual Studio 2010 で、使用言語 C++ である。opencv [10] を用いて実装した。線路の形状認識の流れは下記のようになり、それぞれ図 7 ~ 12 のようになる。

- (1) 前面展望映像をグレー化する
- (2) ガウシアンフィルタで平滑化する
- (3) エッジ抽出する
- (4) テンプレートをグレー化する
- (5) チャンファーマッチングを行う
- (6) コストを比較して線形を判断する

線形を判断した結果は csv (Comma Separated Values) ファイルに書き出すようにした。

3.4.2 考察

テンプレートマッチングでは計算量が多く、認識精度が



図 9 平滑化
Fig. 9 Smoothing

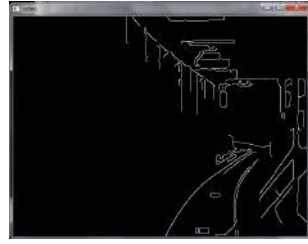


図 10 エッジ抽出
Fig. 10 Edge extraction

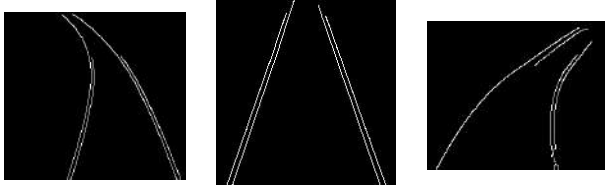


図 11 テンプレート
Fig. 11 Template

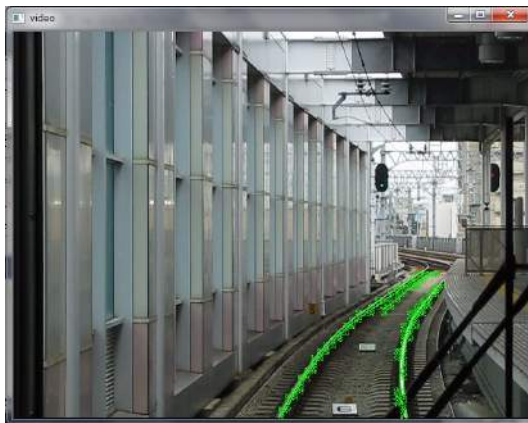


図 12 推定結果
Fig. 12 Estimated result

あまりよくなかった．テンプレートを増やすとさらに計算量が増え，認識精度が低下するという問題があった．そのため，計算量が少なく，認識精度の高い推定方法を用いるべきであることがわかった．

3.5 画面最下段に着目した前面展望映像の線路の形状推定の予備実験

前面展望映像の最下段のピクセルに着目した手法を考え，計算量の削減と認識精度の向上に試みた．図 13 に本手法で注目する線路の中心位置と映像の中心位置の線路の形状による位置関係を示す．図 13 に示すように線路の中心の位置と前面展望映像の中心との位置関係により，線路の形状（直線，右カーブおよび左カーブ）が認識できると考えられる．そこで本研究では，線路の位置を求めるために，実車の前面展望映像をグレー化したのちにエッジ抽出を行う．図 14 にエッジ抽出を行った映像を示す．図 14 に示すように，画面最下段には線路を表す白色のピクセルが表れており，これを用いて線路の中心位置を列挙する．画面

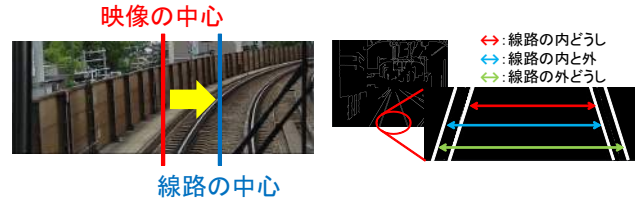


図 13 線路の中心と映像の中心の位置関係

図 14 エッジ抽出した映像と線路の拡大図

Fig. 13 the center of a track and the center of an image

Fig. 14 Edge and Enlargement

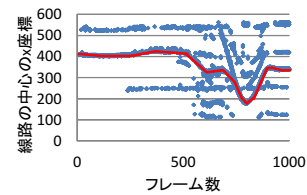


図 15 線路の中心位置候補
Fig. 15 Candidate of the center

最下段には線路を表す白色のピクセルが表れており，これを用いて線路の中心位置を列挙する．理想的な線路は図 14 右側に示すように 4 つの白色ピクセルとそれらの間の距離で表されるが，本研究では，エッジがきれいに取れない場合も考慮し，2 つの白色ピクセル間の距離 d_c が以下の条件 (1) を満たすものを線路として認識し，2 つの白色ピクセルの中間の x 座標を線路の中心位置とする．

$$\begin{aligned} & \min(\text{線路の内側同士の距離が取り得る値}) \\ & \leq d_c \\ & \leq \max(\text{線路の外側同士の距離が取り得る値}) \end{aligned} \quad (1)$$

よって最下段に存在する白色ピクセルの，全ての 2 つの組合せについて d_c を計算することで，そのフレームにおける線路の中心位置の候補を列挙することができる．なお，本研究では， $\min(\text{線路の内側同士の距離が取り得る値})=145$ (pixel)， $\max(\text{線路の外側同士の距離が取り得る値})=180$ (pixel) を基本とし映像によって変更した．

ある実車の前面展望映像について線路幅の候補を 1000 フレームまで列挙した結果を図 15 に示す．縦軸が線路の中心候補の x 座標 (pixel) で横軸がフレーム数である．青が抽出結果で，赤が前面展望映像から取得される正解データである．正解データに分類されない中心候補はワイパーやプラットホームと壁など白ピクセル間の距離が線路幅に近いものであると考えられ，これらを除外して線路の中心位置を絞り込む必要がある．



図 16 システムの外観

Fig. 16 Appearance of system

4. 実装

4.1 システムの外観

システムの外観を図 16 に示す．カメラおよび小型ハイブリッドセンサを搭載した鉄道模型，パソコン，コントローラを用いる．パソコンにはカメラの受信機とセンサが接続されている．

4.2 アプリケーションの実装

アプリケーションの開発環境は Microsoft Visual Studio 2010 で，使用言語は模型の速度および曲率の推定のみ C# で，その他は C++ である．それぞれのプログラムについて説明する．

4.2.1 模型の速さおよび曲率の推定

ハイブリッドセンサの値から，速度および曲率を求める．ハイブリッドセンサから得られる加速度の Y 軸のデータの振幅を用いて，あらかじめ予備実験で求めておいた加速度と速度の関係を用いて速度を求める．速度が求まると，角速度とあわせて曲率を求めることができる．実装したアプリケーションは図 17 のようになり，上から順に加速度 x, y, z ，角速度 x 軸回り， y 軸回り， z 軸回り，速度，曲率を表示している．測定結果は csv ファイルに書き出すようにした．

4.2.2 前面展望映像の線路形状の推定

前面展望映像の線路形状認識を行うアプリケーションは OpenCV[10] を用いて実装した．線路の形状認識の流れは下記ようになる．

- (1) 前面展望映像をグレー化する
- (2) ガウシアンフィルタで平滑化する
- (3) エッジ抽出する
- (4) 線路の中心候補の列挙を行う．
- (5) 線路の中心位置の絞り込みを行う．

予備実験の図 15 の結果より正解データの中心位置候補について，連続するフレーム間ではそれらの x 座標の差は非常に小さいことがわかる．そこで，次のような手順で線路の中心位置を絞り込む．動画のフレーム数を M, m 番

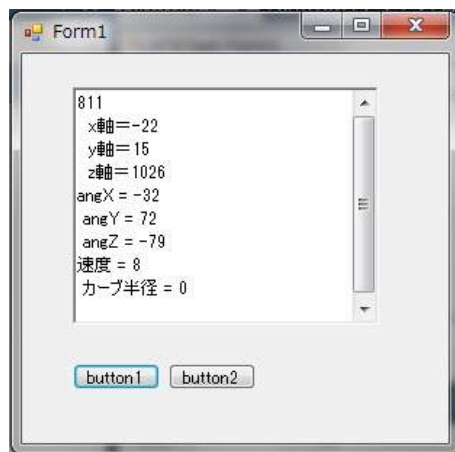


図 17 速度および曲率の推定

Fig. 17 Presume speed and curvature

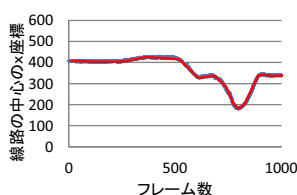


図 18 絞り込みの結果

Fig. 18 Result of narrowing down

目 ($1 \leq m \leq M$) のフレームのある中心位置候補の x 座標を $c(m)$ とし，正解として出力される中心位置 $C(m)$ を式 (2) および式 (3) で定義する．

$$C(1) = (\text{手動で入力}) \quad (2)$$

$$C(m) = \min(c(m) | [|c(m) - C(m-1)| \leq 50]) \quad (3)$$

これを用いて絞り込みを行うと図 18 のようになり，絞り込みを行うことができた．推定結果は csv ファイルに書き出すようにした．

4.3 システムの表示画面

実装のできている部分を用いてシステムを使用すると図 19 のようになる．現在は前面展望映像の列車の速さと模型の列車の速度は合っていない．

5. まとめと今後の課題

本研究では，鉄道模型の線路形状および速さの推定方法について紹介し，実車の前面展望映像の線路認識の手法およびマッチングに DTW を使うことについて提案した．今後は，線路認識については他の手法と認識精度やかかる時間について比較を行う予定である．また，鉄道の特有の構造物 (架線や枕木) を用いた認識手法を考える．さらに，DTW でマッチングを行うことが適当であるのか評価する．ジオラマを認識し，重ね合わせる部分を自動的に変更でき



図 19 表示画面

Fig. 19 A screenshot of the sysytem

るようにする．システムを早期に完成させ，システムの評価実験を行う．

謝辞 本研究の一部は，科学技術振興機構戦略的創造研究推進事業（さきがけ），科学技術振興機構研究成果展開事業スーパークラスプログラム，文部科学省科学研究費補助金基盤研究（A）(23240010) および文部科学省科学研究費補助金挑戦的萌芽研究（25540084）によるものである．ここに記して謝意を表す．

参考文献

- [1] カメラカーで遊ぶ，
<http://www.katomodels.com/hobbycenter/blog/osaka/000434.php>
- [2] 松本典久: DCC で楽しむ鉄道模型, オーム社 (July 2009).
- [3] 東京都交通局総務部経営管理課: 東京都交通局 2013 経営レポート, 入手先
(<http://www.kotsu.metro.tokyo.jp/information/keiei/pdf/report2013.pdf>) (Dec. 2013).
- [4] 小野晋太郎, 松久亮太, 川崎 洋, 池内克史: 車載カメラ映像の時空間マッチングによる自車位置推定, 生産研究, Vol. 63, No2, pp. 209–215 (2011) .
- [5] 木口浩之, 益田泰孝, 渡部広一, 河岡 司: 車載カメラ画像による位置・方位・速度の測定, JSAI2007, 3C9-5 (2007)
- [6] 池ヶ谷直樹, 佐治 斉: 可変テンプレート画像による車両前方のオプティカルフローの検出, 情報処理学会研究報告, 2004-CVIM-144, pp. 87–92 (Apr. 2004) .
- [7] 鶴飼正人, 那須ボクダン友幸, 長峯望: 光学画像センサによる列車前方の監視手法, 鉄道総研報告, Vol. 26, No. 7, pp. 29–34 (Jul. 2012).
- [8] 山内 慎, 西森正志, 西嶋直人, 新谷一也, 松本恭徳, 松井克憲: 列車運転士支援システムの開発 -鉄道模型用標識認識システムの作成-, Journal of Japan Association for College of Technology, Vol. 16, No. 3, pp.35–38 (July 2011).
- [9] N. Cuntoor, A. Kale , and R. Chellappa: *Combining Multiple Evidences for Gait Recognition*, Proc. of IEEE Int. Cont. Acoustics, Speech, and Signal Processing, Vol. 3, pp. 33–36 (Apr. 2003).
- [10] opencv, <http://opencv.org/>