

市街地映像マップと2台の車載カメラを用いた自車位置推定

内山 寛之[†] 出口 大輔[†] 高橋 友和^{††} 井手 一郎[†] 村瀬 洋[†]

[†] 名古屋大学大学院 情報科学研究科 〒464-8601 愛知県名古屋市千種区不老町

^{††} 岐阜聖徳学園大学 経済情報学部 〒500-8288 岐阜県岐阜市中鶉 1-38

E-mail: †{uchiyama,ddeguchi,ide,murase}@murase.m.is.nagoya-u.ac.jp, ††ttakahashi@gifu.shotoku.ac.jp

あらまし 高機能なカーナビゲーションシステムの実現には高精度な位置情報が不可欠であるが、現在の普及型 GPS では進行方向・車線方向のそれぞれで十分な精度が得られない。そこで本研究では、映像と位置情報で構成される市街地映像マップと2台の車載カメラから得られる画像系列を対応付けることで、自車位置推定を行う手法を提案する。あらかじめ、カメラを搭載した車両で街中を走り回り、街並み画像系列に位置情報を付加することでデータベース(市街地映像マップ)を作成する。次に、市街地映像マップと車載カメラ画像系列を DP マッチングにより対応付け、その対応付け結果と2台の通常カメラの撮影方向を用いて自車位置を計算する。実環境での評価実験により、車両の進行方向の位置のみならず、車線方向の位置も推定することができ、提案手法の有効性が確認できた。

キーワード 自車位置推定, 車載カメラ, 市街地映像マップ

Vehicle Position Estimation using Street Image Map and Two In-vehicle Cameras

Hiroyuki UCHIYAMA[†], Daisuke DEGUCHI[†], Tomokazu TAKAHASHI^{††}, Ichiro IDE[†], and
Hiroshi MURASE[†]

[†] Graduate School of Information Science, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya-shi, Aichi 464-8601 Japan

^{††} Faculty of Economics and Information, Gifu Shotoku Gakuen University,
Nakauzura 1-38, Gifu-shi, Gifu 500-8288 Japan

E-mail: †{uchiyama,ddeguchi,ide,murase}@murase.m.is.nagoya-u.ac.jp, ††ttakahashi@gifu.shotoku.ac.jp

Abstract Although accurate ego-localization is required for realizing high-functional car navigation systems, ordinary GPS could not obtain sufficient accuracy. As a solution, we propose an ego-localization method using Street Image Map and two in-vehicle cameras. First, a database (Street Image Map) is constructed by collecting pairs of a streetscape image and its position. Next, two image sequences from two in-vehicle cameras are matched with the image sequence of Street Image Map by using DP matching. Then, the self-position is calculated by triangulation using the matching result and the direction of the two cameras. From experimental results, we confirmed that the proposed method could estimate both lateral and forward position accurately.

Key words ego-localization, in-vehicle camera, street image map

1. はじめに

高精度な自車位置推定は、次世代のカーナビゲーションシステムを実現する上で最も重要な機能の1つである。例えば、カーナビゲーションシステムにおいて、詳細な位置を知ることが、運転者を正確に誘導するために必須である。また、正確な自車位置情報を得ることができれば、走行車線を識別することで、運転者に車線変更を促したり、一時停止すべき箇所ですべて自動的にブレーキを制御する等の運転支援を実現できる。しかし、現在のカーナ

ビゲーションシステムは普及型 GPS によって自車位置を推定しており、建物等による電波の反射や遮蔽のため、3~30m の誤差が含まれる。車速やジャイロスコプ、道路形状などを利用して推定位置の修正が行われるが、上記のような支援を実現するために十分な精度は得られていない。一方、より高精度な RTK-GPS を用いることで数 cm の精度で位置推定を行うことができる。しかし、建物などによる遮蔽に弱いという欠点のほか、現状では高価なため、一般車両に搭載するには課題が多い。これらの問題を解決するために、車載レーザスキャナや車載

カメラなどのセンサを GPS と組み合わせて用いた自転車位置推定手法が検討されている [1], [2]. 特に, 車載カメラはすでに一般車に普及しつつあり, 標識認識 [3] など様々な用途に利用する研究が行われている. そこで我々は, 高精度な自転車位置推定を実現するため, 車載カメラと市街地映像マップ [4] を組み合わせた手法を提案する.

市街地映像マップは, 街並み画像とその位置座標を対応付けたデータベースである. 近年, 記憶装置の大容量化やインターネット回線の高速化に伴って, 商業的に大規模な市街地映像の収集がなされ, それらの情報に容易にアクセスできるようになりつつある. 例えば, Google Maps [5] やロケーションビュー [6] は, 路上の全方位カメラ画像をインターネット上で閲覧できるサービスを提供している. また, 映像などの市街地情報収集を目的として, 高精度に自転車位置推定が可能な機器と, 情報収集の様々なセンサを搭載した専用車両が開発されている [7], [8]. このように, 大量の街並み画像を収集することが可能になりつつあることを踏まえ, 本研究では市街地映像マップをデータベースと利用することで自転車位置推定を実現する.

公道を走行する車両の自転車位置推定は, 移動物の自己位置推定の 1 分野として研究されている. カメラを用いた自転車位置推定は主にランドマークベースの手法とアピランスペースの手法に分けられる. ランドマークベースの手法は, まず環境中に存在するランドマークの 3 次元的位置情報を記憶する. そして, 入力カメラ画像中から検出された複数のランドマークと照合し, それらの位置関係から自己位置を推定する手法である. ここで, ランドマークとは, 環境中のある位置に固定された特徴点である. ランドマークとして, 人工的に設置した目印 [9] の他に, 物体のコーナの位置に関する情報を利用したもの [10] や, SIFT 特徴などの局所記述子を利用したもの [11] など, 自然特徴点を利用したものが挙げられる. 人工的なランドマークは設置のために人的コストがかかるという問題があり, 自然特徴点を利用したものは特徴点の抽出・照合の精度が位置推定の精度に大きく影響するという問題がある.

一方で, アピランスペースの手法では, 物体の見え方を利用する. アピランスペースの位置推定手法では, 画像そのものと, その撮影地点の位置情報を事前に記憶し, 入力カメラから得られた画像に最も類似する画像を探索することで自己位置を得る. これには, 被写体の位置を知る必要がないことや, 特徴点の抽出・照合が不要であるという利点がある. また, この手法では, 時系列情報を利用することで画像の照合精度向上を図るために, DP マッチング [12] を応用した手法が数多く提案されている. 小島らは連続 DP や RIFCDP (Reference Interval-Free Continuous DP) によるロボットの位置同定システムを提案している [13], [14]. 西村らは全方位カメラを用い, Non-monotonic 連続 DP により, ロボット

の向きにかかわらず位置推定を行う手法を提案している [15]. しかし, これらの手法には, 自己の位置はデータベース上に存在するという前提の下で自転車位置推定を行うため, 自己の位置がデータベース上の位置と離れるほど, 位置推定誤差が大きくなるという欠点がある.

以上の問題を解決するため, 本研究では, 方向が既知の 2 台の車載カメラから得られる画像系列と, 市街地映像マップ (データベース) 上の画像系列とを対応付けることにより, 進行方向のみならず, 車線方向の位置をも推定する手法を提案する. 具体的には, 各カメラ画像に最も類似する画像が得られる 2 地点をデータベース上から探索する. 次に, 三角測量を用いて走行位置を計算する. その際, 画像系列間照合により画像間の対応付け精度の向上を図る.

2. 街並み画像系列間の照合による自転車位置推定

2.1 提案手法の概要

提案手法は主に (1) データベース構築と (2) 自転車位置推定の 2 つの段階に分けられる. データベース構築の段階では, まずカメラを搭載した専用車両が街中を走り回ることによって街並み画像とそれに対応する位置座標の組を収集する. この段階では, 広い範囲が一度に撮影できる全方位カメラを使用する. 次に自転車位置推定の段階では, 方向が既知で固定の 2 つの通常カメラから得られた画像に対して, 類似する街並み画像が得られる位置をデータベース上から探索する. そして, その位置関係とカメラの撮影方向から自転車位置を計算する.

図 1 を用いて位置推定の概要を説明する. 自転車位置推定を行う車両が地点 1 にある場合, カメラ 1 から得られる画像はデータベース上の地点 2 から同じ方向を見た街並み画像の一部分に類似したものとなり, カメラ 2 から得られる画像はデータベース上の地点 3 から見た街並み画像の一部分に類似したものとなる. (a) ~ (c) はそれぞれ全方位カメラによる地点 1 ~ 3 の画像の例である. 画像中の矩形は通常カメラの視野とそれに対応する領域をあらわし, 縦軸は光軸を通る鉛直な線を表している. 地点 2, 3 の座標がわかれば, カメラ 1, 2 の角度から三角測量を用いて地点 1 の位置を計算することができる. また, 画像系列は経路に沿って連続に存在することを利用して, 単一の画像対の照合ではなく, 画像系列間照合を用いることで位置推定精度の向上を図る (図 2).

本研究では, カメラの内部パラメータ, 車両上の装着位置, 方向はすべて既知とし, 図 3 の手順に沿って自転車位置を推定する. 以降, 提案手法の処理について詳しく述べる.

2.2 データベース構築

冒頭で紹介したようなデータベース構築用の専用車両を用い, 車両に取り付けた全方位カメラから得られる画

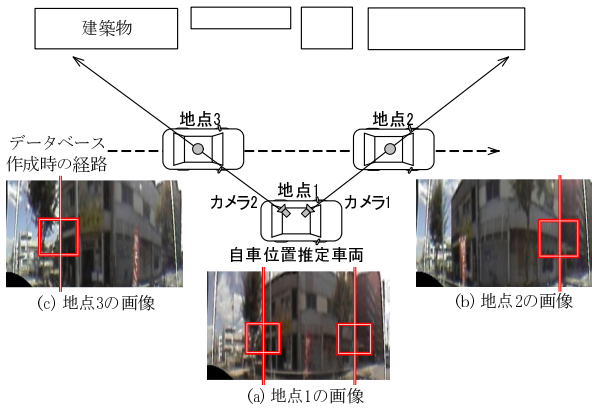


図1 市街地映像マップを用いた自転車位置の三角測量．データベース上の座標とカメラ方向より自転車位置を計算する．

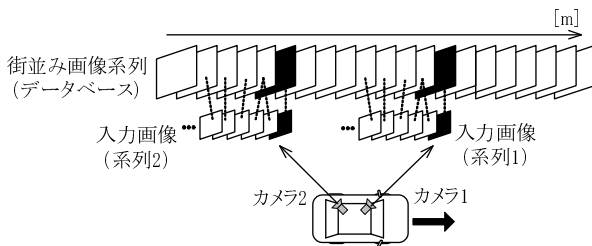


図2 データベース画像系列と入力画像系列の照合．系列間照合により対応付け精度の向上を図る．

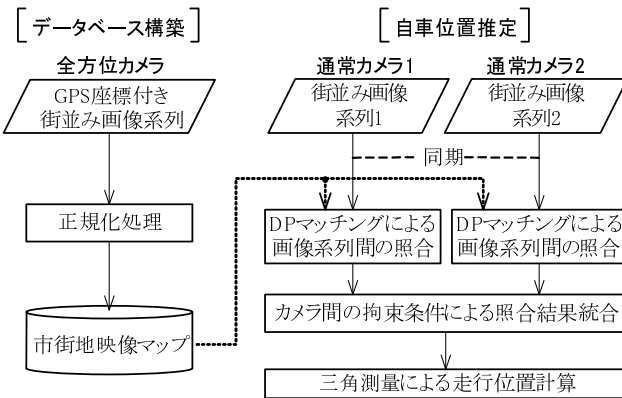


図3 走行位置推定の流れ．提案手法はデータベース構築と自転車位置推定の2段階の処理に分かれる．

像系列と位置情報を同時に取得する．取得した情報により，地図上の位置と画像系列を対応付ける．

次に，フレーム速度の正規化を行う．データベース車両と位置推定車両の極端な速度比が発生した場合には，系列間対応付けの際に問題となる．そのため，後の処理を簡便にする目的で次の正規化を行う．具体的には，画像系列の各フレームが地図上で等間隔になるように，フレームの間引きと挿入によるリサンプリングを行う．このとき，地図上で最近傍のフレームをリサンプリング後のフレームとして選択する．

最後に，カメラの光学的特性を正規化するために，それぞれの映像フレームに対して球面投影を行う．ここで

球面投影は，カメラ中心を球の中心とする球面に画像を投影するものである．

2.3 自転車位置推定

まず，2台のカメラから入力された街並み画像系列に対し，それぞれ独立にDPマッチングを行うことで，データベース内の画像系列と対応付ける．次に，それぞれのカメラの対応付け結果を，2台のカメラ間の拘束条件を用いて統合することで，誤対応の削減を行う．最後に，データベース上の座標とカメラ方向を用いることで，自転車の走行位置を計算する．

2.3.1 DPマッチングによる画像系列間の照合

a) 概要

まず，2台の通常カメラからの入力画像系列に対し，データベース構築時と同様の球面投影を行う．以後，画像照合は球面座標上で行う．

入力画像系列はデータベース作成時の撮影位置と異なるため，画像の見え方が異なる．そこで，弱透視投影を仮定し，カメラを設置する高さの違いや車両の揺れを球面座標上での上下の平行移動で近似する．また，撮影対象物との距離の違いを画像の拡大縮小で近似する．

入力画像系列とデータベース画像系列にはこのように平行移動，拡大縮小の2自由度が存在する．本手法では，街並みの連続性，車両の移動の連続性から，これらのパラメータは連続的に変化すると仮定する．この仮定の下，パラメータの連続性を保持しつつ，データベース画像系列と入力画像系列のフレームの対応付けを行うために，1次元-3次元パターン間のDPマッチングを導入する．1次元-3次元パターン間のDPマッチングは1次元-2次元パターン間のDPマッチング[16]を拡張したものである．これにより，自転車が過去から現在まで獲得した入力画像系列 $0, 1, \dots, \tau_0$ をデータベース画像系列 $0, 1, \dots, t, \dots, T$ と照合する．結果として，現在の入力フレーム τ_0 に対応するデータベース画像系列内のフレームが t であり，座標のインデックスが x ，スケールパラメータのインデックスが s であるときのコスト $g(x, s, t, \tau_0)$ を得る．

また，画像の比較を行う際，撮影機器や撮影環境の違いによる色変化への対応が必要である．ここでは，前処理としてカラーヒストグラムの平坦化[17]により，これに対応する．

b) DPマッチングのアルゴリズム

DPマッチングの具体的なアルゴリズムを説明する．まず，パラメータ x, s のデータベース画像 $I_1(x, s, t)$ と入力画像 $I_2(\tau)$ の距離 $d(x, s, t, \tau)$ として，次式のカラー画像間のSAD (Sum of Absolute Difference)[17]を用いる．

$$d(x, s, t, \tau) = \text{SAD}\{I_1(x, s, t), I_2(\tau)\} \quad (1)$$

ここで， $x = 0, 1, \dots, X - 1, s = 0, 1, \dots, S - 1$ とする．

以下に示す手順により，コスト $g(x, s, t, \tau_0)$ を計算する．

[Step1] Initialization

計算空間の範囲外に ∞ を代入する．

$$\begin{aligned} g(-1, s, t, \tau) &= g(x, -1, t, \tau) \\ &= g(X, s, t, \tau) = g(x, S, t, \tau) \end{aligned} \quad (2)$$

$$= \infty$$

$$g(x, s, -i, \tau) = \infty \quad (i = 1, 2, \dots, t_s) \quad (3)$$

[Step2] Iteration

$\tau = 0, 1, \dots, \tau_0$ について，次の漸化式を計算する．

for $\tau = 0$

$$g(x, s, t, 0) = d(x, s, t, 0) \quad (4)$$

for $\tau = 1, 2, \dots, \tau_0$

$$\begin{aligned} g(x, s, t, \tau) &= \min_{\substack{x_p \in \{-1, 0, 1\} \\ s_p \in \{-1, 0, 1\} \\ t_p \in \{0, \dots, t_s\}}} \{ \\ &g(x - x_p, s - s_p, t - t_p, \tau - 1) \\ &+ w(x_p, s_p, t_p, \tau) d(x, s, t, \tau) \} \end{aligned} \quad (5)$$

[Step3] Output

τ_0 が入力されたときに，パラメータ x, s, t をとるコスト $g(x, s, t, \tau_0)$ を出力する．

式中の $w(x_p, s_p, t_p, \tau)$ は重みである． t_s は時間軸方向に関する傾斜制限であり，位置推定車両の想定される最大速度およびデータベースの 1 フレームあたりの距離を考慮して設定する．具体的には，位置推定車両はデータベース車両の t_s 倍の速度比まで許すことに相当する．また，大規模なデータベースを用いる場合， t の取り得る全ての範囲に対し式 (5) を計算することは記憶容量・計算量の観点から現実的ではない．そこで，整合窓 [18] を導入することで，計算範囲を限定する．

2.3.2 カメラ間の拘束条件を用いた系列間照合結果の統合

カメラ 1, 2 によって推定されるデータベース上のフレームをそれぞれ t_1, t_2 とする．前節の DP マッチングの結果，入力フレーム τ に関する t_1, t_2 の対応付けのパスの候補が無数に存在する (図 4(a))．通常の両端点フリー DP マッチング [12] の枠組みではコストが最小となるパスを最適パスとするが，終端点付近では雑音によって対応付けが不安定になる．そこで，2 台のカメラの拘束条件を利用することで，1 台のカメラで考慮した最適なパスから，2 台のカメラを考慮した上での最適なパスに修正する (図 4(b))．

まず，車両の挙動から，以下の条件を仮定する．

仮定 1 車両は連続的に移動しているため， t_1, t_2 は連続的に変化する

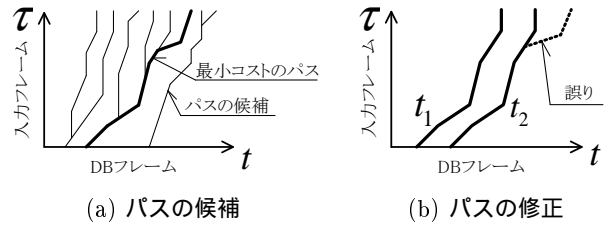


図 4 2 カメラ間の拘束条件による照合結果の統合．条件を満たさないパスを修正する．

仮定 2 車両が進行方向に直進している場合，車線方向の変化は小さく， $t_1 - t_2$ の時間的な変化量も小さい

以上の仮定を満たす (t_1, t_2) の組を探索することにより，誤対応を削減する．まず， τ フレーム目を入力したときにデータベースのフレーム t が選択されるコストを次式で定義する．

$$f(t, \tau) = \min_{x, s} \frac{g(x, s, t, \tau)}{\tau + 1} \quad (6)$$

右辺の分母はコストを入力フレーム数で正規化する項である．以後， $f(t, \tau)$ をカメラ 1, 2 についてそれぞれ $f_1(t_1, \tau)$, $f_2(t_2, \tau)$ と表記する．また，入力画像 τ が入力された時， $t = (t_1, t_2)$ が選択されるコストを次式で定義する．

$$F(t, \tau) = f_1(t_1, \tau) + f_2(t_2, \tau) \quad (7)$$

そして， t_1, t_2, \dots, t_τ を変化させることで，次式を最小化する t_τ を探索する．

$$G(\tau) = \min_{t_1, t_2, \dots, t_\tau} \sum_{i=1}^{\tau} F(t_i, i) \quad (8)$$

ただし，仮定 1, 2 より， t_i の軌跡はほぼ直線とみなせるため， $t_i = (t_{1,i}, t_{2,i})$ の選び方を次式で与える．

$$\begin{aligned} (t_{1,i}, t_{2,i}) \in \{ &(t_{1,i-1} + 0, t_{2,i-1} + 0), \\ &(t_{1,i-1} + 1, t_{2,i-1} + 1), \\ &(t_{1,i-1} + 2, t_{2,i-1} + 2), \\ &(t_{1,i-1} + 3, t_{2,i-1} + 3), \\ &(t_{1,i-1} + 2, t_{2,i-1} + 3), \\ &(t_{1,i-1} + 3, t_{2,i-1} + 2) \} \end{aligned} \quad (9)$$

式 (8) は動的計画法により，効率的に解くことができる [19]．

2.3.3 三角測量による走行位置の計算

2.3.2 の処理により，カメラ 1, 2 に対応するデータベース上の地点 A, B の座標が得られる (図 5)．地点 $A(x_A, y_A)$, $B(x_B, y_B)$ ，カメラの角度 α_1, α_2 ，カメラ間の距離 d_1 を用いて，自車の位置 (x_1, y_1) は次の連立方程式を解くことで得られる．

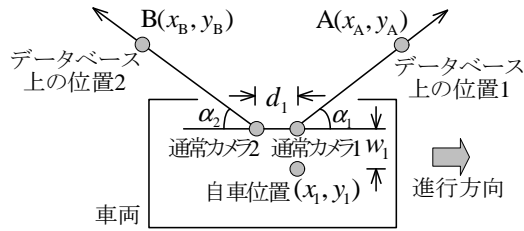


図5 三角測量による走行位置計算．自車位置 (x_1, y_1) はデータベース上の位置 A, B, カメラ方向 α_1, α_2 , カメラの車両上での装着位置を用いて計算する．

$$\begin{cases} y_1 - y_A = (x_1 - x_A) \tan \alpha_1 \\ y_1 - y_B = -(x_1 - x_B - d_1) \tan \alpha_2 \end{cases} \quad (10)$$

また後処理として、車両は車線方向に急激に移動しないと仮定し、車線方向の移動平均をとることで平滑化する．

3. 実験

3.1 実験概要

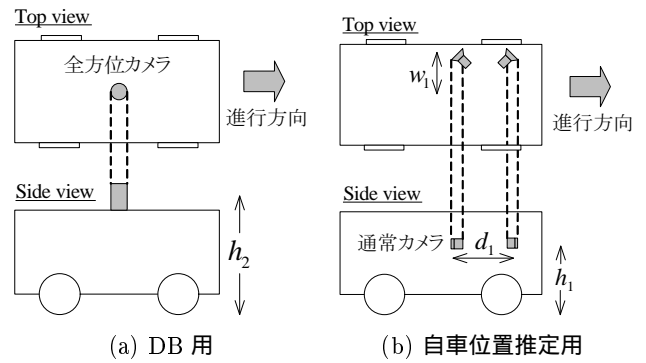
提案手法の有効性を確認するために、走行車線認識率、進行方向・車線方向の位置推定誤差を以下の実験により評価する．比較手法として、提案手法の処理の中で入力画像の平行移動・拡大縮小を行わない手法を用いる．さらに、普及型 GPS 単体の精度との比較を行う．

実験に用いるカメラの車両への装着位置を図6に示す．データベース構築には全方位カメラ VStone 社製 VS-C14N を使用し、路面から $h_2 = 1.5 \text{ m}$ の位置に設置した．自車位置推定用の通常カメラは2台とも民生用の DV カメラで、車内の助手席サイドガラスに固定した．設置位置はいずれも車両の中心線から進行方向に対して左側に $w_1 = 0.5 \text{ m}$ 離れており、2台の通常カメラの間隔は $d_1 = 0.2 \text{ m}$ とした．また、設置高さは $h_1 = 1 \text{ m}$ とした．撮影した映像のフレームレートは、全方位カメラも通常カメラも 30fps であった．

実験に使用したデータセットを表1に示す．経路は片側2車線の直線道路であり、走行中に車両の加減速、停止、車線変更を含む．経路長が 400 m と 300 m の2つの区間を用いて実験を行った．データ収集期間は6か月である．また、データベース用と位置推定用の映像は次の2種類を用いた．

- データベース用と位置推定用の映像を異なる走行で撮影
- データベース用と位置推定用の映像を同じ走行で撮影

前者はデータセット I-A, I-B であり、走行車線認識率、進行方向の位置推定誤差を評価するために使用した．後者はデータセット II-A, II-B であり、データベース構築時と位置推定時で車線方向の位置推定誤差を評価するた



(c) 実験用車両

図6 カメラの装着位置．全方位カメラは車両の屋根に、通常カメラは車内の助手席サイドガラスに固定した．

表1 実験に使用するデータセット

データセット	DB用と位置推定用の映像の撮影タイミング	コース	走行回数
I-A	異なる	A	9
I-B	異なる	B	8
II-A	同じ	A	2
II-B	同じ	B	2

めに使用した．これは、自車の車線内の位置の真値が分かるデータで車線方向の位置推定誤差を評価するためである．この際、1つの車両にデータベース構築用の全方位カメラと位置推定用の通常カメラを装着し、撮影を行った．

データベース構築の際に、画像の位置座標は人手で付与した．提案手法の各パラメータは予備実験により求めたものを使用した．データベース正規化後のフレーム間の位置間隔は 0.4 m であり、式(5)の t_s は3とした．DP マッチングにおける画像の平行移動は 1.44 度刻みで 35 段階、拡大縮小は 0.03 倍刻みで 20 段階とした．重みパラメータ $w(x_p, s_p, t_p, \tau)$ は基本を 1.0 として、 $x_p \neq 0$ または $s_p \neq 0$ のときは 1.1 とした．各評価用のカメラの方向は目視で測定した．内部カメラパラメータはすべて、カタログ値を使用した．入力画像の下半分は道路領域のため、上半分のみを画像照合に使用した．また、実験結果の評価に用いる自車位置の真値は人手で与えた．

3.2 実験結果

以下の実験結果の数値は、停止しているフレームを除き、フレーム単位で算出した．データセット I-A と I-B を用いて測定した進行方向の位置推定誤差が一定範囲に収まった割合を図7に示す．比較のために、入力画像の拡大縮小・平行移動を行わない場合（比較手法）の結果、

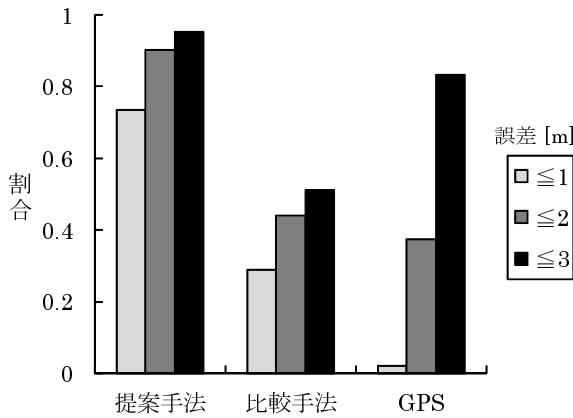


図 7 進行方向の位置推定誤差が一定範囲に収まる割合の比較．比較手法では提案手法の入力画像の平行移動・拡大縮小を行わない場合である．

	位置推定時：左	位置推定時：右
DB 構築時：左	94%	85%
DB 構築時：右	87%	95%

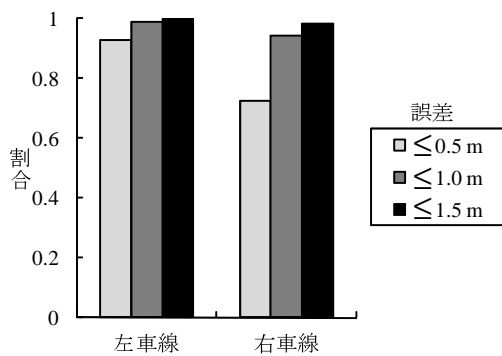


図 8 車線方向の位置推定誤差が一定範囲に収まる割合．データベース構築用の映像と位置推定用の映像は同時に撮影したものを用いた．

および普及型 GPS による結果を並べて示す．次に，走行車線認識率を，データベース構築段階と位置推定段階の車線別に表 2 に示す．車線方向の推定位置の例をいくつか図 9 に示す．グラフの縦軸は進行方向に向かって左が正に対応している．走行車線認識率は，車線幅が 3.0 m かつデータベース構築時に車両は走行車線の中心を走っていると仮定し，車線方向の推定位置が $-4.5 \sim -1.5$ m の場合はデータベース構築時の車線に対して右隣の車線， $-1.5 \sim 1.5$ m の場合は同じ車線， $1.5 \sim 4.5$ m の場合は左隣の車線と判別した．図 9 の点線は走行車線認識に用いたしきい値である．入力画像系列のフレーム数に対する正しい車線の範囲内に入ったフレーム数の割合を認識率とした．データセット II-A と II-B を用いた車線方向の位置推定誤差が一定範囲に収まる割合を車線別に図 8 に示す．

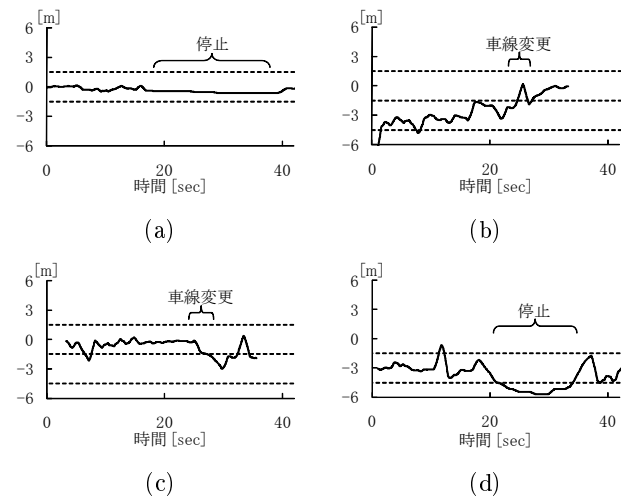


図 9 車線方向の位置推定の例．

4. 考 察

本節では，実験結果に対する考察と，実環境で問題になると考えられる他車両による遮蔽や街並み変化の影響，カメラを設置する高さの違いの影響，時間帯による影響についての考察について述べる．

4.1 実験結果に対する考察

図 7 および表 2 の結果から，良好に自車位置推定を行うことができていることが分かる．また，図 9(b)，(c) のように，車線変更にも対応できた．データセット I-A，I-B を用いた実験により，右車線を走行した場合は左車線を走行した場合に比べて精度が低下するという結果が得られた（図 8）．これは，右車線を走行するほうが，被写体との距離が遠くなり，空間分解能が低下するためだと考えられる．

また，図 7 より，比較手法のように入力画像の拡大縮小・平行移動を行わない場合は位置推定精度が低下することが分かる．探索範囲が固定の場合は，走行する軌道のずれやカメラの高さの違いによって，データベース構築段階と位置推定段階で画像の撮影範囲が異なることがある．そのため，画像間の照合が困難になったと考えられる．既存手法 [14] のように，データベース構築も自車位置推定も視野の狭い通常カメラで行った場合ではこのような問題が発生すると考えられる．一方，提案手法では画像間の対応領域を適応的に探索するため，撮影範囲の変化に対応することができる．

その他に，図 9(a) のように成功する例もあるものの，(d) のように，停止中に正解から徐々にずれていく現象もみられた．これは，停止中は動きがない画像が連続して入力され，単一画像のテンプレートマッチングに近い結果となり，画像照合性能が低下したためと考えられる．この問題の解決方法の 1 つとして，車速を用いた入力画像系列に対するフレーム間隔の正規化が考えられる．

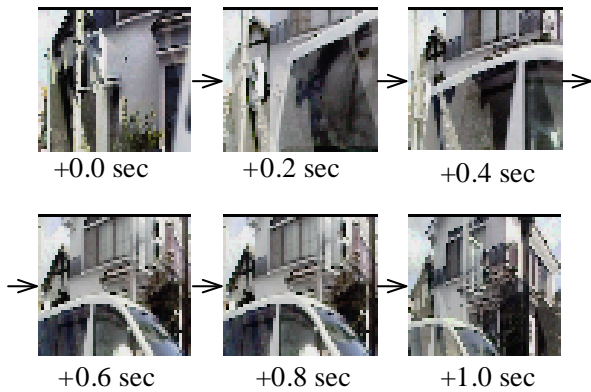


図 10 並走車両による遮蔽．遮蔽が発生すると位置推定誤差が増加する．



図 11 街路樹の落葉による街並み変化．

4.2 他車両による遮蔽や街並み変化の影響

図 10 は他車両を追い抜いた例であるが、このように他車両に視界を遮蔽されたために位置推定精度が低下することがあった．また、図 11 はデータベース構築時には街路樹に葉が茂っていたが、位置推定時には落葉していた例である．このような街並みの経時変化も遮蔽と同様の影響を及ぼすと考えられ、位置推定精度低下の原因となる．その他に、本手法では弱透視投影を仮定し、見え方の違いは画像の平行移動・拡大縮小で近似的に対応したため、撮影位置の違いにより大きく見た目が変化するような部分ではフレーム対応付けの誤りが大きくなった(図 12)．これらのように、画像の照合が困難な状況に対応する方法の 1 つとして、データベース画像と入力画像間の類似度の低下を検出し、その場合はカメラ画像による位置推定結果を採用せず、速度情報から車両の位置推定を行う方法が考えられる．

4.3 カメラを設置する高さの違いの影響

データベース用と自車位置推定用のカメラの高さの差が大きくなると、見え方の変化が位置推定精度に影響すると考えられる．そこで、データベース用のカメラを設置する高さを $h_2 = 2\text{ m}$ に変更して実験を行った．この際、全方位カメラは PointGreyResearch 社製 Ladybug2 を使用し、フレームレート 15fps とした．また、データ



図 12 撮影位置の違いによる見え方の変化．

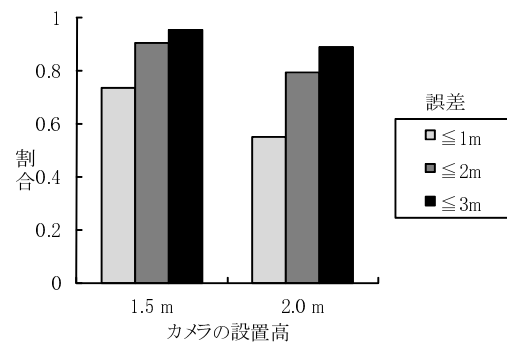


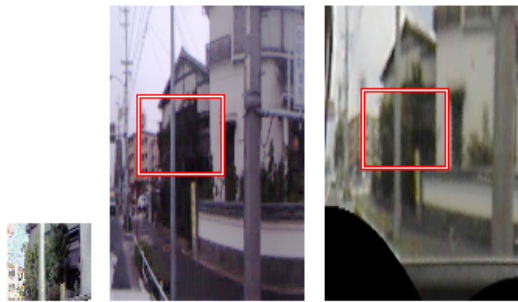
図 13 DB 用のカメラの設置高の違いによる進行方向の位置推定誤差．

ベース構築時には左車線を常に走行した．

進行方向の位置推定誤差をデータベース用のカメラの設置高さ別に図 13 に示す．データベース用のカメラを設置する高さにより結果が異なることが分かる．しかし、対応付け結果の画像を詳しく観察すると、図 12 のように大きく見た目が変化する以外は、大きな位置推定精度の低下は見られなかった．データベース用と自車位置推定用のカメラの高さの差が大きくなるほど、画像の縦方向に見かけ上の伸縮が起きる(図 14)．しかしながら、街並みは横方向に動くため、位置決めには建物の境界などの縦のエッジが重要であり、高さの違いの影響は小さかったと考えられる．今後、カメラを設置する高さの違いと位置推定誤差の関係を詳細に調べる予定である．

4.4 時間帯による影響

本手法は可視光カメラを用いるため、走行する時間帯によっては、照度の不足により本手法の適用が困難になる場合が考えられる．そこで、日没直後に測定した位置推定用のデータを用い、時間帯の違いによる影響を調査した．実験経路としてコース B を選び、1 回走行して測定したデータを用いた．なお、データベース用の映像は日中に撮影したものを使用した．実験の結果、進行方向の位置推定誤差が 2 m 以下に収まった割合は 95% であり、日没直後の時間帯でも提案手法が適用できることが



(a) 入力 (b) データベース (設置高 2.0 m) (c) データベース (設置高 1.0 m)

図 14 カメラを設置する高さの違いによる領域対応付けの違い。すべての画像は同一地点を示している。

わかった。これは、周囲が薄暗いため画像のコントラストは低下するが、建築物の形状は画像上で確認できるためだと考えられる。今後、時間帯による位置推定精度の違いを詳細に調べる必要がある。

4.5 計算時間に関する考察

1 フレームあたりの計算時間は、Core2 Quad Q9550 2.83GHz CPU, 4GB メモリ の環境で 0.5 sec であった。画像探索の初期位置は、普及型 GPS によりおおよその位置を決定すればよいと考えられるため、検索時間はデータベースのサイズに依存しない。また、提案手法の実用化において、必要なデータベースのサイズは、日本の道路の総延長が約 119 万 m であり [20], 1 フレームのサイズが 4k バイト, 1 フレームあたりの間隔が 0.4m として、約 12TB である。これは、1 都道府県当たり平均 253G バイトであり、実用的な数字であると考えられる。以上のことから、本報告で提案した手法の枠組みは、大規模なデータベースにも適用可能であると考えられる。

5. ま と め

2 台の車載カメラから得られる入力画像に対し、類似する画像が得られる 2 地点をデータベース上から探索することによって自車位置を推定する手法を提案した。その際、画像の探索に DP マッチング、走行位置の計算に三角測量を使用した。また、2 台のカメラの拘束条件を用いることにより、誤対応の削減を図った。評価実験の結果、約 9 割の区間で 2 m 以下の誤差で自車位置推定を行うことができ、走行車線認識率は 85 ~ 95 % であった。カメラを設置する高さの違い、時間帯による影響について調べたところ、影響は小さいことがわかった。

今後の課題として、遮蔽に頑強な手法の検討、より長い経路での実験が挙げられる。また、ジャイロスコープや速度計など、他センサとの組み合わせについても検討していきたい。

文 献

[1] 渋谷 奈保, 高橋 友和, 井手 一郎, 村瀬 洋, 小島 祥子, 高

- 橋 新, “車載レーザスキャナによる距離データマップの構築と高精度自車位置推定,” 信学論 (D), Vol.J92-D, No.2, pp.215-225, Feb. 2009.
- [2] 本多 充知, 加藤 博一, 中川 隆志, 山口 喜久, 河合 克哉, 西田 正吾, “低フレームレート画像からの横断歩道検出による自車位置補正,” 信学技報, MVE2007-57, Sep. 2007.
- [3] H. Ishida, T. Takahashi, I. Ide, Y. Mekada, H. Murase, “Generation of Training Data by Degradation Models for Traffic Sign Symbol Recognition,” IEICE Trans., Vol.J90-E, No.8, pp.1134-1141, Aug. 2007.
- [4] 佐藤 准嗣, 高橋 友和, 井手 一郎, 村瀬 洋, “GPS 座標付き全方位映像群からの市街地映像マップの構築と街並変化の検出,” 信学論 (D-II), Vol.J90-D, No. 4, pp. 1085-1095, Apr. 2007.
- [5] Google Maps, <http://maps.google.co.jp>.
- [6] 株式会社ロケーションビュー, <http://www.locaview.com>.
- [7] 山野 芳樹, “モバイルマッピングシステムによる道路空間データ収集手法の開発,” APA, No.77-1, (財)日本測量調査技術会, Nov. 2000.
- [8] 瀧口 純一, 梶原 尚幸, 島 嘉宏, 廣川 類, 黒崎 隆二郎, 目黒 淳一, 石川 貴一郎, 天野 嘉春, 橋詰 匠, “高精度 GPS 移動測位技術を利用した移動監視システムとモバイルマッピングシステムの開発,” 第 25 回日本ロボット学会 学術講演会, Sep. 2007.
- [9] M. Betke and L. Gurfvits, “Mobile Robot Localization Using Landmarks,” IEEE Trans. on Robotics and Automation, Vol.13, No.2, pp.251-263, Apr. 1997.
- [10] E. Royer, J. Bom, M. Dhome, B. Thuillot, M. Lhuillier and F. Marmouton, “Outdoor Autonomous Navigation Using Monocular Vision,” Proc. Int. Conf. on Intelligent Robots and Systems (IROS) 2005, pp.3395-3400, Aug. 2005.
- [11] 薄 充孝, 中川 知香, 佐藤 智和, 横矢 直和, “ランドマークデータベースに基づく静止画像からのカメラ位置・姿勢推定,” 日本バーチャルリアリティ学論, Vol.13, No.2, pp.161-170, June 2008.
- [12] 速水 悟, 岡 隆一, “連続 DP による連続単語認識実験とその考察,” 信学論 (D), Vol. J67-D, No.6, pp.677-684, July 1984.
- [13] 小島 浩, 高橋 勝彦, 関 進, 岡 隆一, “走行ロボット動画からのモーションシーンのスポッティング認識,” 信学技報, PRU93-58, Sep. 1993.
- [14] 小島 浩, 伊藤 慶明, 岡 隆一, “Reference Interval-Free 連続 DP を利用した移動ロボットの時系列画像による位置同定システム,” 信学論 (D-II), Vol.J80-D-2, No. 3, pp. 724-733, Mar. 1997.
- [15] 西村 拓一, 野崎 俊輔, 岡 隆一, “Non-monotonic 連続 DP によるスポッティングに基づく移動ロボットの時系列画像を用いた大局的位置の推定,” 信学論 (D-II), Vol. J80-D-2, No.8, pp. 1876-1884, Aug. 1998.
- [16] H. Sakoe and Y. Katayama, “One Dimensional-Two Dimensional Dynamic Programming Matching Algorithm for Character Recognition,” IEICE Trans. Information and Systems, Vol.E77-D, No.9, pp.1047-1054, Sep. 1994.
- [17] 田村 秀行, “コンピュータ画像処理,” オーム社, 2002.
- [18] 内田 誠一, “DP マッチング概説 ~基本と様々な拡張~, ” 信学技報, PRMU2006-166, Dec. 2006.
- [19] R. ベルマン, S. ドレイファス, “応用ダイナミック・プログラミング,” 日本科学技術連盟, 1962.
- [20] 国土交通省道路局, “道路統計年報 2007,” Nov. 2007.