時系列圧縮 Gist シーン特徴に基づく視覚自己位置推定

近藤賢祐† 田中完爾† 池田剛一郎† 鈴木貴之† 角谷崇徳†
関島正平†

† 福井大学工学部 〒 910–8507 福井市文京 3-9-1
E-mail: †{kondo,ikeda,suzuki,kadoya,sekijima}@rc.his.u-fukui.ac.jp, ††tnkknj@u-fukui.ac.jp

あらまし 視覚移動ロボットの自己位置推定問題は，視覚特徴（ランドマーク）の地図上で自己位置を推定することを目的とする．本論文は，ヒトのシーン知覚特性に着想を得た，圧縮 Gist シーン特徴と呼ぶ，普遍的かつコンパクトな，新しいランドマークを提案する．自動車ロボットの自己位置推定実験により，推定性能，顕著性，情報圧縮などの観点から，提案方法の有効性を検証する．

キーワード 視覚移動ロボット，圧縮 Gist シーン特徴

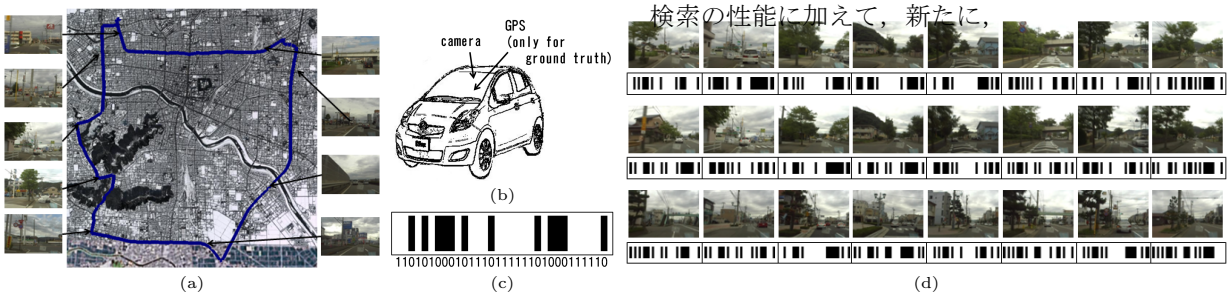


図 1 How well compact binary landmark representation works in mobile robot localization?

(a) Experimental environment and robot's trajectory. (b) A high-speed car-like mobile robot. (c) Binary landmark representation using the semantic hashing technique. (d) Sequences of images and binary codes. Top, Middle: Two similar locations. Bottom: A dissimilar location. Note that the codes are similar/dissimilar only at a few bits.

1. はじめに

本論文では，ランドマークを軽量かつ意味的等価な表現へ翻訳する，ランドマーク要約問題を考える．一般に，自律移動ロボットの自己位置推定タスクは，ロボットがランドマーク地図を用いて，自己位置を一意に推定することを目的とする [1]．そのために，ロボットは，移動経路上の各地点において，ランドマークを認識し，これと類似する特徴を地図中から検索することで，次第に自己位置を絞り込んでいく．従来，この認識・検索の性能に優れる様々なランドマーク（e.g. SIFT, view sequence, color histogram）の研究開発がなされてきた．近年，SLAM 技術の進展により，自律移動ロボットのセンサデータ群をもとに大規模環境のランドマーク地図をリアルタイムに生成することが可能になってきた（e.g. ”大規模 SLAM [2]”）．さらに，この技術を基盤として，不特定多数のロボットが互いの地図を共有利用する自律分散ネットワークの研究がなされている（e.g. ”ロボットセンサネットワーク [3]”）．それに伴い，上記の認識・検索の性能に加えて，新たに，

- 普遍性：普遍的であり，様々な作業環境（例：都市，自然）に有効であること
- 軽量性：コンパクトであり，記憶・送受信に有効であること

という二つの要求を満たす新しいランドマーク技術が望まれる．以上を踏まえ，圧縮 Gist シーン特徴と呼ぶ，新しいランドマーク技術を開発することを本研究の目的とする．

本研究では，上記要求を満たすものとして，シーンの Gist に注目した [1]．一般に，ヒトの視覚システムは，シーンの空間表現を瞬時に獲得することができる．この空間表現は，シーンの要点（Gist 特徴）と呼ばれ，たとえば，シーンの意味（例：道路がある），主要な物体（例：道路の両側に高い壁がある），大域的な構造（例：視野の広がり）など，シーンに関する豊富な情報を含む [4]．これは，Gist 特徴のコンパクト性および普遍性を表してい

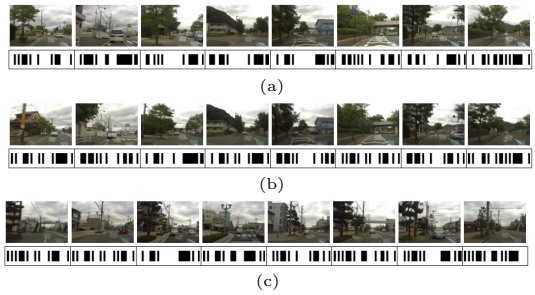


図 2　Compressed Gist sequence.

る．近年，コンピュータビジョンの分野において，この Gist シーン特徴を画像処理技術として工学的に実装する試みがなされている [5]- [8]．Oliva ら [9] は，画像の低空間周波数成分を抽出するフィルタを用いて，Gist シーン記述子を開発した．文献 [10] では，多層グラフィカルモデルに基づく次元削減技術，セマンティックハッシング (SH) [11] を利用して，この Gist シーン記述子を圧縮し，コンパクトな 32bit の 2 値表現，圧縮 Gist へ変換する方法を開発した．この Gist および圧縮 Gist は，近年，画像補完 [8] や画像検索 [10] などの応用において，最先端の認識性能・検索性能を達成している．

本実験では，実験プラットホーム（図 1）として，視覚センサを搭載した自家用車（図 1(b)）を利用し，街中の約 20km の道路（図 1(a)）を，0-40km/h の速度で走行し，自己位置推定を行う（図 1）．多くの既存研究 [12] のような，内界センサや GPS などの位置計測センサを前提としない．事前に，各々の視点について，$k = 32$ ビット圧縮 Gist（図 1(c)）を記録し地図とする．この地図に基づき，標準的なパーティクルフィルタ（モンテカルロ自己位置推定）を用いて推論を行う．本実験を通して，各々のシーン特徴の出現頻度，ランドマークとしての顕著性，従来法 [13] との性能比較など，各種の調査を行う．さらに，ビットマスクを用いて，圧縮 Gist の冗長なビットを間引き，さらなる情報圧縮を試みる．その上で，ビット数と推定性能の関係を調べる．

1.1 関 連 研 究

本研究の特色は，ロボットがナビゲーション中に取得する時系列圧縮 Gist を用いる点にある [1]．Gist を認識タスクに用いた研究事例として，上記の画像補完 [8] や画像検索 [10] があるが，これらは，単一画像のみに基づいて認識を行っていた．本研究のように，時系列圧縮 Gist を用いることで，単独の圧縮 Gist で得られる情報量（高々32 ビット）よりも多くの情報量を利用することが可能となる．また，時系列圧縮 Gist の持つ冗長な情報量を利用して，さらなるコンパクト性の向上を期待できる．以上の観点から，時系列圧縮 Gist に基づく自己位置推定システムを提案することを，本論文の目的とする．

既存の自己位置推定システムは，トラッキングに基づく方法と，場所認識に基づく方法とに大別できる．前者は，自己位置の事前知識を利用し，保持・追跡する点に特色がある．代表例として，Davison [14] らの拡張カルマンフィルタに基づく monoSLAM や，Klein らによるキーフレームに基づく PTAM [15] などがある．本論文においても，パーティクルフィルタに基づくトラッキング方法を用いる．一方，後者は，自己位置の事前知識を必ずしも必要とせず，視覚画像と類似する場所を地図中から検索することで自己位置を推定する．例えば, [16] では，structure-from-motion による三次元地図生成，任意視点画像生成，および，類似画像検索を利用し，広域自己位置推定を実現している．一方, [17] では，全方位カメラと三次元 LIDAR による精密な広域地図生成システム（および公開データセット）と，安価なカメラによる汎用の自己位置推定システムの組合せを研究している．以上の研究事例を踏まえた上で，本論文では，時系列圧縮 Gist シーン特徴に基づく普遍的・軽量なランドマークを研究する．

2. ランドマークの表現と利用

Gist シーン記述子 [9] は，シーン画像を入力とし，知覚次元と呼ばれる，自然性（naturalness），開放性（openness），粗野性（roughness），拡張性（expansion），起伏性（ruggedness）などの大域的な特徴を捉え，シーンの空間的な構造を記述する．そのために，画像のスペクトルや粗い位置情報を利用して特徴抽出を行う．具体的には，異なる複数のスケールについて，方向フィルタを施し，画像分割を行い，4x4 のグリッド上で，各々の出力ベクトルの大きさを平均する．本実験では，方向数 8，スケール数 4 の方向フィルタを用いて，(4x4)x8x4=512 次元の特徴ベクトルを得る．

この Gist 特徴を，セマンティックハッシング技術 [11] により，コンパクトな 32bit ビット列（圧縮 Gist）へ翻訳する．図 2(a) に，ある走行経路を移動したときの圧縮 Gist 列を示す．また，図 2(b) および図 2(c) は，それぞれ，同じ走行経路を再度移動したとき，および，異なる走行経路を移動したときの圧縮 Gist 列である．図からも分かるように，同じ走行経路で観測された圧縮 Gist 列は，互いに類似している部分が多い．ただし，完全に一致してはいない．また，単一の圧縮 Gist だけでは，自己位置を一意に決定できない．このため，複数の圧縮 Gist を手がかりとし，自己位置推定を行うことが不可欠といえる．

セマンティックハッシング（以下，SH）は，多層グラフィカルモデルに基づく機械学習手法である [18]．この多層モデルは，下層から上層に進むにつれてノード数が大幅に減少していくピラミッド構造となっている．この SH 関数を用いて，Gist シーン記述子を圧縮し，k [bit] の圧縮 Gist

$$c = [c^1, ..., c^k] \tag{1}$$

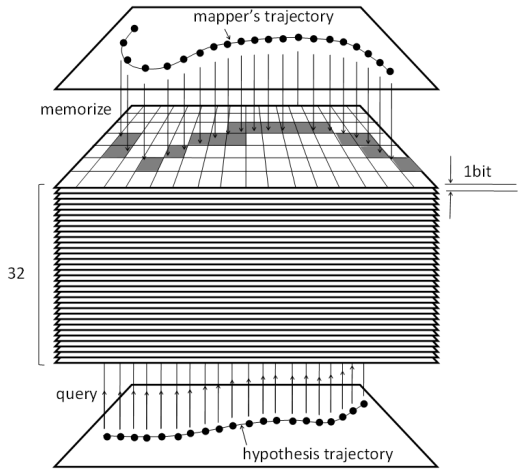


図 3　Binary landmark map.

を得る．各々のビット c^i を，独立した 2 値観測 $c^1, ..., c^k$ ととらえ，k 枚の地図へ記録する．図 3 に示すように，各々の 2 値地図は，画像（視点）当り 1bit を消費する．この空間コストは，使用する 2 値地図の枚数に比例する．既に述べたように，本論文の実験では，$k = 32$ 枚の 2 値地図すべてを使用するケース，および，δk[bit] を間引いた k'（$k' = k - \delta k < k$）枚の 2 値地図を使用するケースの 2 通りについて，性能評価を行う．

本章の残りの部分では，自己位置推定タスクにおいて，この 2 値地図を利用する方法を説明する．一般に，自己位置推定アルゴリズム（例：パーティクルフィルタ，複数仮説追跡）は，自己位置について複数の仮説を生成し，ロボットによるオドメトリおよびランドマークの観測をもとに，各仮説の追跡および尤度評価を行う．我々のアプローチでは，尤度評価と仮説生成の 2 つの処理部分において 2 値ランドマークを用いる．

2.1　尤 度 評 価

尤度評価は，ある自己位置の仮説 x_t に対し，ランドマークの観測尤度を評価することを目的とする．そのために，ロボットが x_t にいるという仮定の下で，ランドマーク位置を算出し，同じ位置にあるランドマークを地図中から検索し，2 つのランドマーク間で圧縮 Gist の各ビット c^i を比較する．そして，地図上での当該ビット $c_{map,x}$ との比較に基づいて，ビット毎の観測尤度

$$P(c^i|x) \simeq \begin{cases} l_o c_L^{1/k} & (c^i = c_{map,x}) \\ l_o & (c^i \neq c_{map,x}) \end{cases} \tag{2}$$

を算出する．そして，すべてのビットについての観測尤度を掛け合わせることで，ランドマークの観測尤度を得る．c_L は，観測尤度の大きさを表す．この c_L は，k 個の全てのビットについての観測値が地図と合致したとき，それらの観測尤度を全て掛け合わせたものに等しい大きさを持つ．

本研究では，この合致判定のために，ビットカウント操作を利用する．ビットカウント操作とは，あるアドレス対 a, b を入力とし，a, b 間のハミング距離が一定値以下かどうかを高速に判定する方法である．

2.2　仮 説 生 成

仮説生成は，最新の観測特徴に基づいて，自己位置の新しい仮説を生成することを目的とする．そのために，地図データベースの中から，類似特徴を検索し，その類似特徴に対応する視点位置を求め，これを新しい仮説とする．この類似検索は，質問画像と参照画像のビット列の間のハミング距離が閾値以下となる参照画像を高速に検索することを目的とする．

本研究では，高速な類似検索のために，ハッシュテーブルを利用する．本ハッシュテーブルにおいて，2 値コード $C = [c^1, \cdots, c^k]$ は，アドレス

$$a = \sum_i^k c^i 2^{i-1} \bmod S \tag{3}$$

を指す．S は，ハッシュテーブルのサイズであり，実装では，8MByte に設定した．地図データベースに新しいランドマークを挿入する処理は，ランドマークのポインタをアドレス a にある該当ビンに挿入する処理となる．この挿入処理は，典型的な地図生成タスクにおいて [19]，新たなランドマークを地図に追加する処理と並列に，逐次的に行うことができる．ハッシュテーブル参照を高速化する手段として，あるアドレス a を入力とし，そのアドレスを中心とするハミングボール内の全ての相対アドレスを返すような参照テーブルを利用する．

3.　自己位置推定システム

自己位置推定の処理は，視覚画像列から自己移動量を推定する（視覚オドメトリ），視覚画像から Gist シーン記述子により視覚特徴を認識する（特徴抽出），視覚特徴をセマンティックハッシングにより次元削減し圧縮 Gist を算出する（圧縮），および，自己移動量と圧縮 Gist をもとにモンテカルロ自己位置推定（パーティクルフィルタ）を用いて推定値を更新する（推論），という各処理部分からなる．このうち，特徴抽出および圧縮の処理については，既に説明した．以下では，視覚オドメトリ（3.1）および推論（3.2）について説明する．

3.1　視覚オドメトリ

視覚オドメトリは，連続する画像列を入力とし，自己移動量を推定することを目的とする．素朴な方法として，局所特徴の追跡に基づく方法（例：monoSLAM）やキーフレームの照合に基づく方法（例：PTAM）などがある．しかし，これらは，追跡・照合する局所特徴やキーフレームが存在することを前提にしている．本実験のように，ロボットが高速移動する場合，観測地点の間隔が大きい，ロボット本体が高速振動する，などの理由により，これ

らの方法を用いることができない．この点を踏まえ，本研究の視覚オドメトリは，ロボットが移動しているかどうかを検出するという単純な問題を扱う．その手段として，オプティカルフローを用いる．まず，現在および一つ前のフレーム画像からなる画像対を入力としオプティカルフローを算出し，すべてのフローについてベクトル長を算出する．そして，それらの中間値が閾値長さを越えるかどうかを判定する．その結果を，ロボットが移動している (1) か否 (0) かを表す 2 値データ（自己移動量）とする．

3.2 推　　論

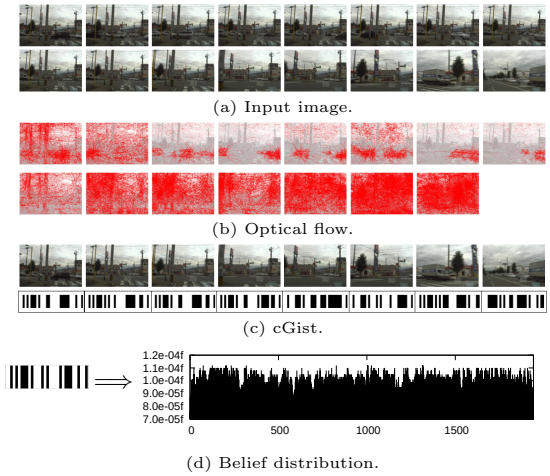
モンテカルロ自己位置推定 [12] は，自己移動量および視覚特徴の観測列をもとに，自己位置の信念分布を更新する．各時刻の信念分布は，自己位置の候補（サンプル）の分布として，離散的に表現する．ただし，各サンプルは，その確からしさを表す重み，および，自己位置の情報を持つ．まず，自己位置推定タスクの開始時に，自己位置の空間全体に一様重みのサンプル群を一様に分布させ，初期のサンプル集合とする．そして，自己移動量および特徴観測のデータを取得する毎に，それぞれ，運動更新および知覚更新と呼ばれる処理を実行する．運動更新は，自己移動量に基づいて，各サンプルの位置を移動する処理である．本実装では，自己移動量が 1（移動）の場合，各サンプルについて移動量を抽出し自己位置に加算する，0（静止）の場合，どのサンプルの自己位置も変更しない，という処理になる．一方，知覚更新は，観測特徴と地図中の各特徴との類似度に基づいて，2.1 で述べた方法により，各サンプルの尤度を更新する処理である．また，有効サンプル数 [20] が閾値よりも小さいとき，リサンプリング処理により，サンプルの再配置を行う．また，大域自己位置推定やエラーリカバリの手段として，センサリセット処理 [21] を導入し，毎回のランドマーク観測において，サンプル群の一部分（全体の 1%）をランダムに選択し，2.2 で述べた方法により，知覚モデルに従う新しいサンプルへ置き換える．

4. 評 価 実 験

4.1 設　　定

屋外において自己位置推定実験を行い提案方法の有効性を検証する．実験プラットホームとして，単眼カメラを搭載した自家用車（以下，ロボット）を用いる（図 1）．一般に，自家用車の自己位置推定タスクは，視点間隔が大きい，カメラが高速振動する，などの理由から，挑戦的な課題と考えられている．

本実験では，主に 3 種のデータセットを用いる．第一に，LabelMe ウェブサイト [22] からダウンロードした 70,000 枚（以下，”LabelMe”と呼ぶ）の画像を，セマンティックハッシングを学習するための訓練用データとして用いる．第二に，ロボットが図 1 の道路を時速 0-40km で走行した際の画像列（”mapping”）を，地図生成タスクのための観測データとして用いる．その際，車載 GPS により，ロボットの自己位置を取得し，地図中のランドマーク位置として用いる．第三に，ロボットが図 1 の道路を再度走行した際の画像列（”localization”）を，自己位置推定タスクのための観測データとして用いる．その際，車載 GPS によりロボットの自己位置を取得し，自己位置推定の正解データとして用いる．また，”mapping”と”localization” は，異なる日時に取得し，天候はいずれも晴天であった．信号停止などの理由により，ロボットが長時間停止することがあり，その時区間のデータは削除した．フレームレートは，10fps であった．たとえば，ロボットが時速 40km で走行したとき，視点間隔は，約 1m となる．



(a) Input image.
(b) Optical flow.
(c) cGist.
(d) Belief distribution.

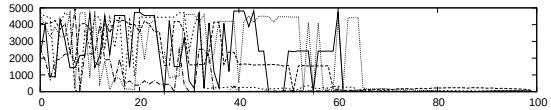
図 4　Self-localization process.

特に断りがない場合，パーティクルフィルタのサンプル数を 10,000，観測尤度の大きさを $c_L = 2$，ビット数を $k = 32$ とした．また，ランドマークやロボットの位置を表すために，道路に沿って，一次元の位置座標系を導入した．また，圧縮 Gist および自己位置推定システムの実装には，C++ 言語を用いた．セマンティックハッシングを実装する際，公開 Matlab コード [10] を参考にした．図 4 に自己位置推定処理の様子を示す．

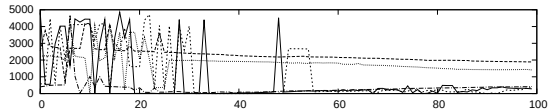
4.2 実験方法と結果

4.2.1 推 定 性 能

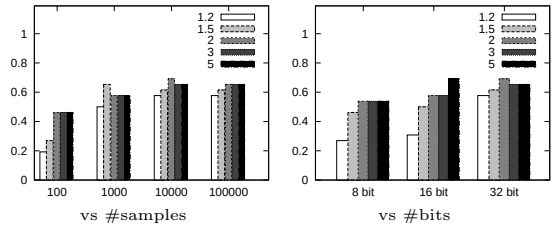
まず，自己位置推定の性能を調べた．そのために，移動経路長 100 視点に相当する自己位置推定タスクを，100 回実施し，成功割合を求める．各タスクの移動経路は，”localization” 上でスタート地点とゴール地点の対により定める．まず，スタート地点をランダムに定め，つぎに，移動経路長が 100 視点となるようにゴール地点を定める．各々の移動経路について，自己位置推定タスクを実施し，最終的に推定結果が収束したものを対象として，推定の成功割合を求める．また，最終的な推定誤差が 200m（経路長の 1%に相当）よりも小さくなった場



(a) Success examples.

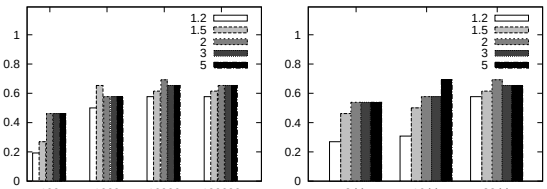


(b) Failure examples.



(c) Success rate.

図 5　Localization accuracy.
Localization errors for 5 success examples (top) and for 5 failure examples (bottom), randomly sampled from the 100 localization tasks. Vertical axis: location error [m]. Horizontal axis: location ID. Successful task is defined as such tasks where the localization error finally becomes smaller than 200m, about 1% of the entire trajectory length. The length of localization trajectory corresponds to 100 viewpoints.



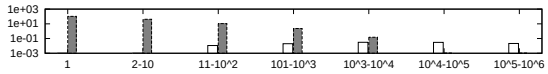
(a) vs #samples　(b) vs #bits

図 6　Success rate.

合を，推定の成功とし，それ以外を，失敗と定義する．

図 5 に，自己位置推定タスクの成功例と失敗例について，推定誤差が時間変化している様子を示す．図中のグラフで，横軸は視点数を，縦軸は推定誤差 [m] を，それぞれ表す．各々の例は，100 通りの自己位置推定タスクの中から，無作為に 5 つを選び出したものである．自己位置推定タスクの開始時には，自己位置は完全に未知であり，推定誤差は，非常に大きい．成功例の場合，この推定誤差が徐々に減少していき，最終的には，ほぼ零へ収束した．この場合，自己位置推定タスクにおいて観測された視覚特徴の外見が，地図中で同一地点にある視覚特徴の外見と合致し，その結果として，パーティクルフィルタにおいて，正しい仮説が高い尤度を与えられた．一方，失敗例の場合，推定誤差は，全く収束しないか，あるいは，十分に小さくならないか，のいずれかであった．

図 6(a) に成功割合を調べた結果を示す．ただし，図中の各々のプロットは，パーティクルフィルタの重みパラメータ c_L の各々の値に対応する．図より，パーティクルフィルタのサンプル数 10,000 以上の場合，ほぼ 0.7



Left:test data. Right:”LabelMe”.

図 7　Word frequency.

程度の高い成功割合が得られている．また，図 6(b) に，圧縮 Gist のビット数 k（8bit, 16bit, 32bit）と成功割合の関係を調べた結果を示す．32bit の場合に最も安定した結果が得られている．

4.2.2　汎 化 性 能

本実験におけるセマンティックハッシングの汎化能力について考察する．セマンティックハッシングは，訓練データ（”LabelMe”）だけでなく，未知のテストデータに対しても，高い汎化能力を示すことが知られている [11]．一般に，訓練データとテストデータとの差異が大きいほど，高い汎化能力が要求される．この差異を可視化するために，本実験で使用した”mapping” および ”localization” の和集合をテストデータとし，テストデータおよび訓練データ（”LabelMe”）のそれぞれについて，視覚特徴の出現頻度ヒストグラムを求める．

図 7 に結果を示す．図は，テストデータにおいて，全ての視覚特徴を出現頻度の順にソートし，出現頻度に基づいて 7 つのグループへ分類し，各グループについて，テストデータおよび LabelMe 内での出現回数を集計したものである．図より，訓練データとテストデータの間には，大きな差異があることが分かる．たとえば，テストデータでは出現頻度が上位 10 に入っている視覚特徴は，いずれも，”LabelMe” には全く出現していない．このように，出現頻度に大きなギャップがあるにもかかわらず，提案方法は，自己位置推定に成功していることが分かる．

4.2.3　顕　著　性

個々の視覚特徴の顕著性について考察する．一般に，全ての視覚特徴が同じ頻度で観測される訳ではなく，特徴間で顕著性に違いがある．視覚特徴の顕著性は，ランドマーク選択（landmark selection）やランドマーク学習（landmark learning）などの応用において，視覚特徴の重要度を評価する上で重要になる．本実験では，個々の視覚特徴の出現頻度が顕著性を表している．そこで，個々の視覚特徴について，出現頻度を調べる．

図 8 に結果を示す．ここでも，図 7 と同様に，テストデータを 7 つのグループへ分類し，グループごとに出現回数を集計した．参考のため，各グループに対応する入力画像を示す．また，各グループごとに，4 つのプロットがあるが，それぞれ，左から，当該単語，および，その 1bit 近傍，2bit 近傍，3bit 近傍の単語を表す．

個々の視覚特徴の出現頻度には大きな相違があることが分かる．たとえば，最頻出の 100 の視覚特徴は，その他の視覚特徴と比べて，10 倍から 100 倍の高い出現頻

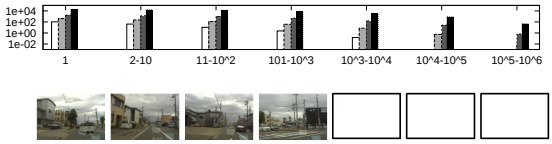
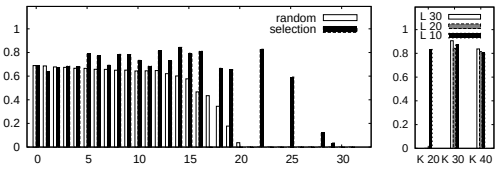


図 8　Word histogram.
Frequency of visual words. Top: Average number of landmarks per word. Words are sorted in terms of the number of landmarks they contain and then grouped into 7 groups. Each datapoint from left to right respectively corresponds to the word as well as their near neighbors in terms of Hamming distance 1, 2 and 3. Bottom: Images corresponding to the central words (ID: 1, 5, 50 and 500) of each group. There is a large difference of frequency between individual visual words. For instance, the top 100 most frequent words are 10-100 times more frequent than the other words. Small number of such frequent words correspond to the most useful landmarks in our system.



(a) SH　(b) LSH

図 9　Localization performance.
(a) The semantic hashing localization. Left: The random sampling strategy. Right: The planned sampling strategy. Vertical axis: success ratio. Horizontal axis: the number ΔK of removed binary maps. (b) The LSH-based localization [13]. In some settings (e.g. {ΔK:22,"random"}, {L:20,K:20}), the success ratio is unreliable (due to low convergence rate) and omitted.

度となった．すなわち，少数の高頻出特徴が重要な特徴の大部分を占めていることが分かる．この結果は，圧縮 Gist 地図をさらに要約できる可能性があることを示唆している．

4.2.4　情 報 圧 縮

次に，2. でも述べたように，ビットマスクを用いて，より少ないビット数 $k' = 32 - \Delta k$ の 2 値地図を用いて自己位置推定タスクを実施し，その推定性能を調べる．ここでは，単純に，Δk 枚の 2 値地図をランダムに間引くことを考える．

図 9 (a)"random" に，2 値地図の数 Δk を変化させながら，成功割合を調べた結果を示す．図より，削減数 Δk が 12 程度以上になると，自己位置推定の成功割合が極端に低くなっていることが分かる．その原因として，ビット数を削減することにより，視覚特徴に含まれていた，有用な情報が除去されたことが考えられる．また，推定誤差が収束しないケースも多くみられた．図中で，データが欠けている箇所が，それらのケースに該当する．

4.2.5　地 図 選 択

次に，4.2.4 のように，ビットをランダムに間引くのではなく，有用度に基づいて間引く情報圧縮方法を考える．これは，ビットマスクの複数の候補の中から，有用なものを選び出す処理となる．その手段として，シミュレーション経験 [23] のアプローチが有効である．すなわち，まず，計算機シミュレーションを利用して，仮想的な自己位置推定タスクを実施し，ビットマスクの有用度を評価する．そして，有用度が最も高かったビットマスクを採用する．その際，ビットマスクの候補数は，ビット数に対し指数関数的に増大するので，比較的少数の候補をランダムに生成し用いることにする．また，仮想的な自己位置推定タスクのための観測データとして，"mapping" や"localization" とは独立に，検証用データセットを取得しておく．また，100 通りの移動経路について，仮想的な自己位置推定タスクを実施し，それらの推定誤差の平均値を求め，その逆数を有用度として用いる．以上の方法を計画サンプリングと呼ぶ．

図 9(a)"selection" に，成功割合を調べた結果を示す．ランダムサンプリング方法とは対照的に，ビット削減にともなう成功割合の減少は緩やかであることがわかる．その傾向は，ビット削減数 Δk が 12-15 程度に至るまで続く．特に，ビット削減数 Δk が 5-15 程度の場合，32 bit の 2 値地図を用いた場合よりも，高い推定性能となっている．これは，元々の 32 bit 地図に含まれていた，不適切・誤りの情報を，シミュレーションを通して，検出・除去できたことによる．

ビットマスクを用いることにより，特徴当りのビット数を節約することができる．たとえば，12 bit を間引く場合，各々の特徴は 32-12=20 bit となる．

4.2.6　従来法 [13] との比較

LSH (Locality Sensitive Hashing [24]) に基づく自己位置推定（LSH localization [13]）を比較手法とし，性能比較を行う．この比較手法は，筆者らが [13] で提案した手法であり，次元削減技術として，セマンティックハッシングではなく，LSH を用いる．本実験において，LSH は，Gist シーン記述子の 512 次元ベクトルを入力とし，場所 ID を出力する．なお, [13] では，ランドマークとして，局所特徴（shape context 記述子）を用いているが，本論文の LSH localization では，大域特徴（Gist シーン記述子）を用いるという違いがある．

図 9(b)"LSH" に結果を示す．図中において，K および L は，LSH の次元数およびハッシュテーブル数であり，それぞれ，値が大きいほど，誤検出および検出漏れを抑制する効果が大きくなる．図より，セマンティックハッシングは，広い範囲のパラメータについて，LSH と同等の性能を示している．

パラメータ設定に依存して，LSH の方が SH よりも高い性能を示しているケースがある．その理由として，LSH は，Gist シーン記述子を場所 ID へ直接に写像するため，汎化誤差の影響を受けないことが挙げられる．その反面，LSH は，特徴当り 40Byte の大きな空間コストがかかるという課題がある．

これに対し，提案方法の特色は，特徴当りの空間コストが非常に低い点にある．提案方法の空間コストは，主

に，セマンティックハッシング（多層グラフィカルモデル）および視覚特徴集合からなる．視覚特徴集合のコストは，特徴数および特徴当りのビット数に比例する．セマンティックハッシングのコストは，入力ベクトルの自乗オーダである．本実験において，上記のコストは，8KB（特徴当り 32 bit，2,000 特徴）および 5.3 MB （特徴数によらず一定）であり，識別的かつコンパクトなランドマーク地図を得ることができた．

5. むすび

本論文では，普遍性および軽量性を特色とする，時系列圧縮 Gist に基づく自己位置推定システムを提案した．提案方法は，Gist シーン記述子を，セマンティックハッシングを用いて 32bit のビット列へ圧縮し，コンパクトな視覚特徴として利用する．実験において，全 32 bit を用いるケース，および，より少ないビット数（例：20 bit）を用いるケース，の両方において，圧縮 Gist が自己位置推定に有効であることが示された．今後，大規模地図の共有・利用を伴う，長期地図学習（long-term map learning）や情報共有ネットワーク（information sharing networks）などのアプリケーションにおいて，本研究のような，普遍的かつ軽量なランドマーク技術は，重要な役割を担うと考える．

6. 謝　辞

本研究は，H21-22 文部科学省科学研究費補助金「移動ロボットによる高次元視覚センサを用いた大規模地図生成」の一環として実施した．本研究の一部は，倉田財団倉田奨励金，および，立石科学技術振興財団研究助成の支援を受けた．

文　献

[1] K. Ikeda and K. Tanaka. Visual robot localization using compact binary landmarks. In *Proc. IEEE Int. Conf. Robotics and Automation*, 2010.

[2] Viorela Ila Kai Ni Frank Dellaert, Justin Carlson and Charles E. Thorpe. Subgraph-preconditioned conjugate gradients for large scale slam. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2010.

[3] Arthur Martens Rene Iser and Friedrich M. Wahl. Localization of mobile robots using incremental local maps. In *Proc. IEEE Int. Conf. Robotics and Automation*, 2010.

[4] A. Torralba. How many pixels make an image? *Visual Neuroscience*, 26:123–131, 2009.

[5] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *Int. J. Computer Vision*, 42(3):145–175, 2001.

[6] J. Hays and A.A. Efros. Im2gps: estimating geographic information from a single image. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, 2008.

[7] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proc. Int. Conf. Image and Video Retrieval*, 2009.

[8] James Hays and Alexei A Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (SIGGRAPH 2007)*, 26(3), 2007.

[9] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *Int. J. computer vision*, 42(3):145–175, 2001.

[10] A. Torralba, R. Fergus, and Y. Weiss. Small codes and large image databases for recognition. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, pages 1–6, 2008.

[11] R. Salakhutdinov and G. Hinton. Semantic hashing. *Int. J. Approximate Reasoning*, 2008.

[12] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 1322–1328, 1999.

[13] K. Tanaka and E. Kondo. A scalable algorithm for monte carlo localization using an incremental e2lsh-database of high dimensional features. *Proc. IEEE Int. Conf. Robotics and Automation*, pages 2784–2791, 2008.

[14] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29:2007, 2007.

[15] Georg Klein and David Murray. Parallel tracking and mapping on a camera phone. In *Proc. Eigth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'09)*, Orlando, October 2009.

[16] A. Irschara, C. Zach, J.M. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. pages 2599–2606, 2009.

[17] Gaurav Pandey, James McBride, Silvio Savarese, and Ryan Eustice. Extrinsic calibration of a 3d laser scanner and an omnidirectional camera. In *7th IFAC Symposium on Intelligent Autonomous Vehicles*, volume 7, 2010.

[18] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504–507, 2006.

[19] M. Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem with Unknown Data Association.* PhD thesis, Carnegie Mellon University, 2003.

[20] A. Doucet, N. Freitas, and N. Gordon editors. Sequential monte carlo methods in practice. *Statistics for engineering and information science*, 2001.

[21] S. Lenser and M. Velose. Sensor resetting localization for poorly modeled mobile robots. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 1225–1232, 2002.

[22] B. Russell, A. Torralba, and W. T. Freeman. Labelme: The open annotation tool. *http://labelme.csail.mit.edu/*.

[23] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson. Landmark selection for vision-based navigation. *Trans. IEEE Robotics*, 22:334–349, 2006.

[24] A. Andoni, M. Datar, N. Immorlica, P. Indyk, and V. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. *Nearest Neighbor Methods in Learning and Vision: Theory and Practice*, 2006.