Vol.2011-CG-142 No.1
2011/2/8

GPU と方向マップに基づく局所不変特徴量のオンライン抽出

市　村　直　幸[†1]

近年，局所不変特徴量は，画像の対応付けや物体認識における基盤要素として幅広く用いられている．局所不変特徴量は，(1) 局所領域の設定，(2) 記述子の計算，の 2 段階の処理を通じて抽出される．本論文では，これらの特徴抽出に必要な数多くの局所演算を，GPU(Graphics Processing Unit) による並列処理を用い高速化することを目的する．特に，輝度勾配の方向ヒストグラムに基づく記述子の計算において，方向マップと呼ばれるデータ構造を導入する．その結果，高速な処理，および，特徴数の変動に対する計算時間の変動が小さいという性質を有する特徴抽出が実現できることを示す．この性質は，高速な処理，および，シーン内容によらずフレームレートを一定に保つことが望ましいオンライン処理において有用である．IEEE1394 カメラおよび NVIDIA GeForce GTX480 を用いたシステムでの実験により，方向マップの導入の有用性を確認する．

On-the-fly Extraction of Local Invariant Features with GPU and Orientation Maps

Naoyuki Ichimura[†1]

Local invariant features have been widely used as fundamental elements for image matching and object recognition. Local invariant features can be extracted by the following two steps: (i) detecting local regions, (ii) calculating descriptors. The purpose of this paper is to consider fast computational techniques for extracting local invariant features through the parallel computing by a graphics processing unit (GPU). In particular, the data structure called *orientation maps* is introduced into the computations of local descriptors consisting of the histograms of gradient orientations. We show that the introduction of orientation maps enable us to realize feature extraction with the properties of the fast computations and the stability of computational times in terms of the changes in the number of features. The properties are useful for on-the-fly extraction in which fast computations and the stability of frame rates are required regardless of the contents of scenes. The usefulness of orientation maps is demonstrated by the experiments using the system with IEEE1394 cameras and the GPU, NVIDIA GeForce GTX 480.

1. まえがき

局所不変特徴量は，画像の特徴表現の一種である．この特徴量は，(1) 画像内での局所領域の設定，(2) 局所領域の画像特徴を表す記述子 (descriptor) の計算，の 2 段階の処理を通じて抽出される[1)–4)]．図 1 に局所領域の例を示す．図中の正方形が局所領域を表す．このような局所領域で計算された記述子は，局所領域内部の輝度やテクスチャ，エッジ等に基づいて画像特徴を数値化したものであり，多くの場合，ベクトルの形態をとる．

局所不変特徴量には，主として 2 つの利点がある．1 つは，局所領域を用いることによる，視野逸脱や遮蔽による隠れへの耐性である．シーンの一部に隠れが生じても，見えている局所領域の特徴量が使用できる．もう 1 つの利点は，その名が示す通り，特徴量に不変性を付与できることである．スケールスペースピラミッドの利用や局所座標系の導入等を通じ，画像の幾何学的変換や輝度変化に対し特徴量が不変になるように，上記 (1),(2) の特徴抽出処理を構成できる．これらの利点にから，隠れや視点の移動，照明条件等の違いにより，基準画像からの見えの変化がシーンにおいて生じたとしても，その変化の影響を軽減し，シーンから基準画像と同様の特徴量を得ることができる．そのため，局所不変特徴量は，画像の対応付けや物体認識の基盤要素として幅広く用いられている[1)–16)]．

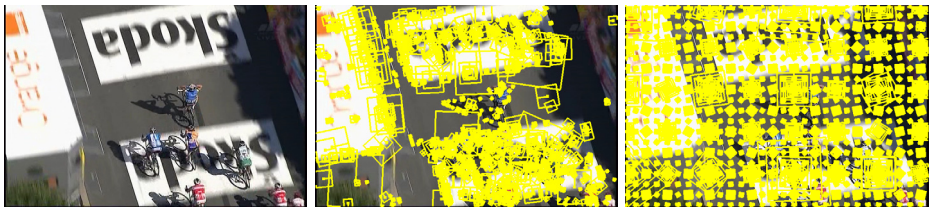
特徴量に不変性を付与する処理の方針には，次の二つのものがあると考えられる．一つ目は，想定される見えの変化を入力画像に生じさせ，その結果から特徴抽出を行うことである．二つ目は，見えの変化量を推定し，その推定結果に基づいて変化した見えを基準画像の状態に変換することである．前者の例として，LoG (Laplacian of Gaussian) フィルタによるスケールスペースピラミッドの生成が挙げられる．入力画像に対し，複数回のダウンサンプリングを適用する．そして，その結果の画像に対し，種々の分散を有する LoG フィルタを適用し画像内容の構造解析を行う．この処理は，認識対象の大きさの変化を想定していることになる．よって，そのスケールスペースピラミッドから抽出される特徴量には，スケール不変性が付与される．後者の例としては，dominant orientation(DO) を用いた処理[2)] が挙げられる．この処理では，局所領域で輝度勾配の方向ヒストグラムを作成し，そのピークを求める．そのピークに対する輝度勾配の方向が DO である．この DO は局所領域内部の形状情報により定まる．よって，画像が回転しても局所領域内部の形状に変化が生じない限り，DO は局所領域で設定した局所座標系においては変化しない．このことから，DO を用いて画像の回転角を推定し，回転による変化を補正することで，基準となる状態での特徴量が計算できる．結果として，特徴量には回転不変性が付与される．

実際の局所不変特徴量の抽出においては，上記の二つの方針を併用している場合が多く，

†1 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology (AIST)

© 2011 Information Processing Society of Japan

Vol.2011-CG-142 No.1
2011/2/8



(a) (b) (c)

図 1 異なる方法による局所領域の設定結果．(a) 原画像”Tour de France”[17]．(b) 特徴点抽出とエッジサンプリングによる方法[15]．(c) グリッドサンプリングによる方法．対応付けや物体認識等，用途に応じて局所領域の設定方法は組み合わされる．ここでは，表示の簡明さのため，全局所領域の一部のみを表示している．

スケールスペースピラミッドの生成やヒストグラムの計算等において，数多くの局所演算が必要となる．本論文では，それらの特徴抽出における数多くの局所演算を，GPU(Graphics Processing Unit) による並列処理を用い高速化することを目的とする．特に，輝度勾配の方向ヒストグラムに基づく記述子の計算を高速化することを検討する．

記述子の計算を高速化することの必要性は，特徴抽出における計算量を前述の 2 段階の処理に分けて考えると明らかとなる．局所領域の設定では，ガウシアンやラプラシアン，微分等のフィルタにおける畳み込み演算が処理の主体となるため，画像の解像度で計算量はほぼ定まる．一方，記述子の計算では，図 1 に示すような局所領域を数多く設定するほど処理対象となる画素数が増加していく．特に，最近の研究[7)–12),14)–16)] から，画像の対応付けや物体認識において，密 (dense) に局所領域を設定することの有効性が示されている．この密な局所領域の設定により，記述子の計算において処理対象となる画素数が画像の解像度を大幅に越えることが多くなり，計算負荷が重くなるのである．

本論文では，輝度勾配の方向ヒストグラムに基づく記述子に対する計算量を削減するため，GPU による並列処理に，Tola ら[16)] により提案された方向マップと呼ばれるデータ構造を導入する．そして，方向マップを導入した特徴抽出では，従来の記述子の計算方法よりも高速であることに加え，特徴数の変動に対する計算時間の変動が小さくなることを示す．これらのことは，高速な処理，および，シーン内容によらずフレームレートを一定に保つことが望ましいオンライン処理において有用である．IEEE1394 カメラおよび NVIDIA GeForce GTX480 を用いたシステムでの実験により，方向マップの導入の有用性を確認する．

2. 輝度勾配の方向ヒストグラムに基づく記述子

本節では，処理の高速化の対象とする，輝度勾配の方向ヒストグラムに基づく記述子について説明する．輝度勾配の方向ヒストグラムに基づく記述子には，SIFT[2)]，GLOH[3)], HOG[7)],

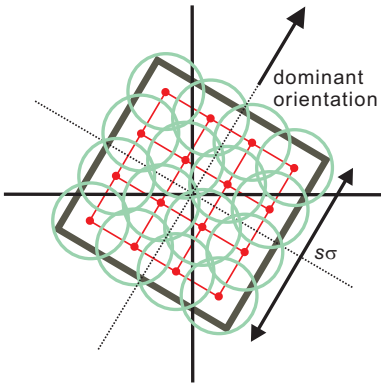


図 2 正方形の局所領域内部におけるセルの配置．局所領域の大きさは，中心画素の固有スケール σ にスケール係数 s を乗じて決定する．この図では，局所領域に 4×4 のセルを配置している．小さな丸はセルの中心を，円はセルの大きさを表す．特徴量は，各セル内部のヒストグラムを連結して構成される．

DAISY[16)] 等があるが，基本部分には，以下に述べる処理が含まれている．

まず，多重解像度解析を行うために，入力画像からスケールスペースピラミッドを生成する．局所領域の設定に特徴点を使用する場合には，ダウンサンプリングと LoG フィルタ等を組合せ，スケールスペースピラミッドを生成する．そして，スケールスペースでの極値点を探索し，特徴点の位置と固有スケール[2),18)] を得る．その特徴点の位置を中心とし，固有スケールに比例する大きさを有する局所領域を画像内に設定する．局所領域の設定には，特徴点以外にエッジ等も利用される[5),9),14),15)]．図 1(b) は，特徴点とエッジを併用した局所領域の設定例である．また，輝度勾配が記述子において必要とされるため，ダウンサンプリングとガウシアンフィルタ，微分フィルタを組合せて別途スケールスペースピラミッドを生成する．このスケールスペースピラミッドにおいて，局所領域の固有スケールに対応するエッジ画像を用い，記述子を計算する．

次に，記述子の計算方法を説明する．記述子の計算では，まず，図 2 に示すように局所領域にいくつかのセルを配置する．各セル内部の画素における輝度勾配の方向を得て，そのヒストグラムを生成する．この際，輝度勾配の方向に対応するビンには，セル内部の画素のもつ輝度勾配の大きさに基づいた値を投票する．そして，全セルの輝度勾配の方向ヒストグラムを連結したベクトルを構成する．このベクトルに対しノルムの正規化を行ったものが，その局所領域の記述子となる．輝度勾配の方向はエッジの方向と関連があることから，この記述子が表現しているものは，セルの配置と大きさに依存した形状情報と言える．このような形状情報が，画像の対応付けや物体認識において有用であることは，多くの研究で確認さ

© 2011 Information Processing Society of Japan

Vol.2011-CG-142 No.1
2011/2/8

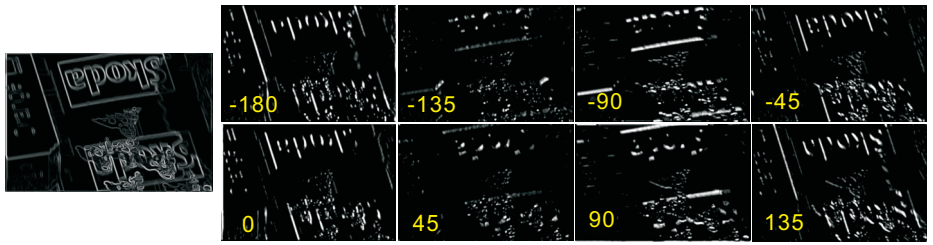


図 3 方向マップの例．この例では，左に示す 1 枚の微分フィルタの処理結果から，右に示す 8 つの輝度勾配の方向に対応する方向マップを生成している．各方向マップは，輝度勾配の大きさを保持している．

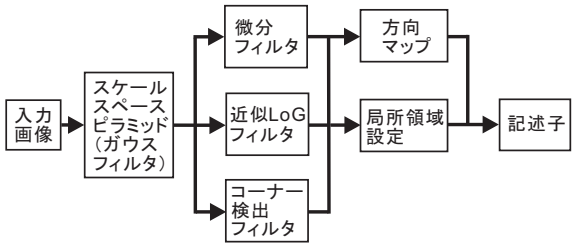


図 4 局所不変特徴量抽出処理のフローチャート．特徴点抽出および密なエッジサンプリングにより，局所領域を設定する[15]．記述子は，多重解像度方向マップを用いて計算する．

れている[1)–16)]．

3. 多重解像度方向マップを用いた記述子の計算

スケールスペースピラミッドの全画素での輝度勾配の方向と大きさは，スケールスペースピラミッドに微分フィルタを適用し，多重解像度エッジ画像が生成された時点で計算できる．この段階で，各画素の輝度勾配の方向がヒストグラムのどのビンに投票されるかが定まる．このことから，各エッジ画像に対して，ヒストグラムのビンに対応する複数の 2 次元配列を用意し，離散化された輝度勾配の方向毎に輝度勾配の大きさを保持することができる．このヒストグラムのビン対応する 2 次元配列を方向マップと呼ぶ[16)]．図 3 は，その方向マップの例である．

方向マップに対してガウシアンフィルタを適用することは，フィルタのスケールに応じた範囲で，輝度勾配の方向の発生頻度を輝度勾配の大きさとガウス関数により定まる重み付きで求めることに相当する．よって，フィルタのスケールを記述子を求める際に用いるセルの大きさに相当するように設定すれば，フィルタリングはセル内の画素によるヒストグラムへの投票値の計算と等価になる．よって，畳み込み済みの方向マップが得られれば，そのマップにおいて各セルの中心位置での値を参照するだけで，輝度勾配の方向ヒストグラムが得られる．つまり，従来の方法のように，セルの全画素を参照する必要がなくなる．

方向マップを用いる方法における問題点は，多重解像度エッジ画像から生成される数多くの多重解像度方向マップに対し，ガウシアンフィルタを適用する必要があることである．多重解像度方向マップの数は，スケールスペースピラミッドのオクターブ数 (ダウンサンプリングの回数)，各オクターブのスケール数，および，輝度勾配の方向ヒストグラムのビン数により定まる．例えば，オクターブ数を 5，スケール数を 3 とし，ビン数を図 3 のように 8 とすると，多重解像度方向マップの数は 120 枚となる．ダウンサンプリングにより解像度は減少していくものの，このような多数の方向マップに対してガウシアンフィルタを適用する計算量は少ないものとは言えない．よって，このガウシアンフィルタの計算が方向マップを用いない場合の記述子の計算よりも効率良く実行できるかどうかが，方向マップの導入により特徴抽出の計算時間を短縮する際の重要な問題となる．

GPU では，メモリレイテンシの短い共有メモリを活用し，効率良くガウシアンフィルタを実装できる．よって，GPU と方向マップの導入は，記述子の計算時間の短縮に有効であると考えられる．次節では，その有効性を確認するための特徴抽出の実装を説明する．

4. GPU による実装

本節では CUDA(Compute Unified Device Architecture)[19)] を用いた特徴抽出の実装について述べる．CUDA では，処理対象となるデータをいくつかのブロックに分け，それぞれのブロックに複数のスレッドを付随させる．ブロック数とスレッド数は execution configuration(以下，EC と記す) と呼ばれる．ブロック数は，ブロックの幅や高さにより間接的に指定する場合が多い．

図 4 は，実装を行った処理のフローチャートである．以下に，この図には記していない補助的な処理も含め，その実装内容を示す．特に，GPU 内部のメモリと EC の適切な選択は高速な計算を実現するために重要であるので，具体的なメモリの種類と EC の値を示す．メモリの使用方法のより詳細に関しては，文献[15)] を参照されたい．

Image transfer: ホストコンピュータ (CPU) から GPU への画像データの転送．

Y Component: カラー画像からの輝度画像生成．ブロックサイズは 16×32．ブロックの全画素にスレッドを割り当て並列化．

Down sampling: スケールスペースピラミッドの生成に必要な画像のダウンサンプリング．EC は **Y Component** と同じ．

Gaussian filter: ガウシアンフィルタを用い，スケールスペースピラミッドを生成．初期スケールは 1.6．フィルタサイズは，ガウス関数の値が 10^{-3} 未満になる点で定義域を打

© 2011 Information Processing Society of Japan

ち切って決定．等方ガウス関数の変数分離性を利用．行処理における 1 次元のブロックサイズは 128 とし，全画素にスレッドを割り当てる．列処理でのブロックサイズは 48×16 とし，ブロックを高さ 8 のサブブロックに分ける．1 つのサブブロックの全画素にスレッドを割り当て，各スレッドで他のサブブロックの同一位置にある画素に対し逐次処理を行う．サブブロックを用いて処理を行うのは，1 つのブロックにおけるスレッド数に上限が存在するためである．

Gradient filter: 5×5 の微分フィルタ[20] による輝度勾配の計算．ブロックサイズは 16×16 とし，ブロックを高さ 8 のサブブロックに分ける．そして，**Gaussian filter** の列処理と同様にスレッドを割り当てる．

ALoG-CD filter: コーナー検出フィルタを併用する近似 LoG フィルタ[21),22)] による，特徴点抽出のためのスケールスペースピラミッドの生成．EC は **Gradient filter** と同じ．

Feature point: **ALoG-CD filter** で得られるスケールスペースにおける 3×3×3 近傍での極値探索による，特徴点およびその固有スケールの抽出．各オクターブ毎に抽出．近似 LoG フィルタとコーナー検出フィルタの応答に対するしきい値処理により，特徴点を選択．各しきい値は，10 および 100．ブロックサイズは 16×32 とし，全画素に対しスレッドを割り当てる．得られた特徴点の位置と固有スケールを，局所領域の位置とスケールとする．

Edge sampling: 微分フィルタの処理結果の全スケールにおいて，微分フィルタの応答がしきい値以上，かつ，空間 3×3 近傍の極大点となる画素をサンプリング[14),15)]．微分フィルタの応答のしきい値は 10．サンプリングした画素の位置とスケールを，局所領域の位置とスケールとする．EC は **Feature point** と同じ．

Orientation map: 多重解像度方向マップの生成．微分フィルタの出力から，図 3 に示すように輝度勾配の方向を 8 つに離散化した方向マップを生成．EC は **Feature point** と同じ．図 2 でのセルの大きさに相当する，多重解像度方向マップへ適用するガウシアンフィルタのスケールは，付録 A.1 の方法で設定．フィルタリング処理は，**Gaussian filter** と同じ．

Dominant orientation: 回転不変性を特徴量に付与するための dominant orientation[2)] の計算．局所領域の大きさは，局所領域のスケールの 5 倍．16 個の局所領域をブロックとし，局所領域を単位として並列化．ブロック内の全局所領域にスレッドを割り当てる．Dominant orientation を求めるためのヒストグラムの生成には，記述子の計算と同様に多重解像度方向マップを利用した．

Descriptor: 記述子の計算．局所領域の大きさは，図 2 のスケール係数 s を 20 として決定した．EC は **Dominant orientation** と同じ．図 2 に示すように，局所領域に 4×4 のセルを配置．多重解像度方向マップを用いた処理では，各セルの中心位置でのみ畳み込み後の方向マップの値を参照し，輝度勾配の方向ヒストグラムを生成．方向マップを用いない場合には，各セル内部の全画素を参照．回転不変性付与のため，dominant orientation に基づきヒストグラムを並行移動．全セルのヒストグラムを連結後，ノルムの正規化を行い記述子を得る．

5. 実験結果

本節では，4 節で示した処理を CPU および GPU により実装し，計算時間を計測した結果を示す．実験には，以下の計算機環境を使用した；ホストコンピュータ: サードウェーブ社 Fractica SE，OS: Fedora13，CPU: Intel Core i7 XE-980X (3.33GHz)，メモリ: 24GB DDR3，グラフィックスカード: NVIDIA Quadro 5000，および，GeForce GTX 480．ディスプレイへの表示は Quadro で行った．演算専用とした GeForce は 480 個のコアを有し，Compute Capability[23)] は 2.0，1 つのブロックにおけるスレッド数の上限は 1024 である．CPU では単一コアを使用して特徴抽出を行った．浮動小数点演算は，全て単精度で行った．

5.1 方向マップの導入による高速化について

まず，図 1 に示した画像”Tour de France”，および，feature detector evaluation sequence[24)] の”Graffiti” の 1 枚目の画像 (図 6 に写っている画像) を用いて，計算時間の計測を行った．画像”Tour de France” の解像度は 720×480，”Graffiti” の解像度は QVGA サイズに縮小した．また，それぞれの画像に対し，スケールスペースピラミッドのオクターブ数は 5 および 4 とした．また，各オクターブにおけるスケール数は 3 とした．輝度勾配の方向は 8 方向で離散化したので，120 枚および 96 枚の多重解像度方向マップが生成された．

計算時間の計測結果を表 1 に示す．表 1 の CPU と GPU は，方向マップを用いた実装を示す．また，GPU–C は，GPU による方向マップ用いない実装を表す．“Orientation map”，“Dominant orientation” および “Descriptor” において GPU と GPU–C を比較することにより，方向マップが記述子の計算時間の短縮に有用であることが確認できる．また，GPU により，各画像において，対 CPU 比で 34 倍および 20 倍程度の特徴抽出の高速化が達成できることも明らかとなった．

5.2 オンライン処理における計算時間の変動について

方向マップを用いた記述子の計算方法では，畳み込み済みの方向マップにおいてセルの中心位置の値を参照するだけで，輝度勾配の方向ヒストグラムが得られる．このように，セル内の全画素を参照する方向マップを用いない方法と比較し，方向マップを用いる方法では各セルでの処理が簡便である．よって，抽出される特徴数の変動に対し，計算時間の変動が小さくなるものと考えられる．この点を確認するために，図 1(c) に示すようなグリッドサンプリングによる局所領域の設定方法を用い，グリッド点数の変化に対する計算時間の変化を計測した．その結果が図 5 である．この図より，特徴数の変動に対する計算時間の変動が，

© 2011 Information Processing Society of Japan

表 1 局所不変特徴量抽出に必要な計算時間．単位は [ms]．OS(Linux) の非リアルタイム性を考慮し，100 回の処理の平均値を示す．総計算時間には，ここに示したタスク以外の処理，例えばメモリアロケーション等も含まれる．

Image	Tour de France			Graffiti		
Image size	720×480			320×240		
Task/Implement	CPU	GPU	GPU–C	CPU	GPU	GPU–C
Image transfer	N/A	0.727	0.725	N/A	0.171	0.171
Y component	2.899	0.089	0.086	0.561	0.055	0.055
Down sampling	0.678	0.142	0.132	0.123	0.086	0.084
Gaussian filter	222.428	4.785	4.653	44.562	2.109	2.149
ALoG-CD filter	241.255	6.078	6.008	48.297	1.850	1.840
Gradient filter	162.334	2.976	2.952	32.810	0.940	0.940
Feature point	130.206	5.065	4.932	28.305	1.263	1.277
Edge sampling	5.901	5.311	5.157	2.034	1.257	1.265
Orientation map	1089.790	24.581	N/A	217.490	10.573	N/A
Dominant orientation	41.546	1.630	4.328	29.831	0.828	2.502
Descriptor	32.700	2.799	51.459	13.175	1.552	29.371
Total	1930.409	55.927	82.055	417.403	21.543	40.516
#features	5656			2647		

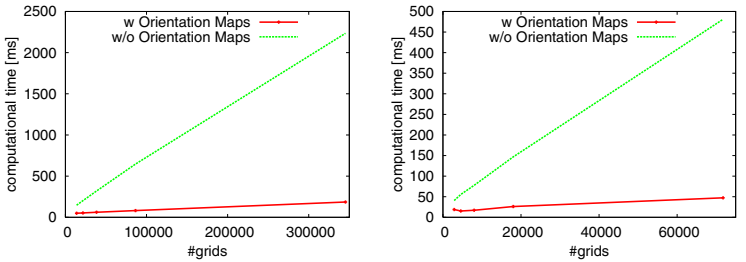


図 5 グリッドサンプリングにより得た特徴の数に対する計算時間の変化．(a) Tour de France. (b) Graffiti. 方向マップを用いることにより，特徴数の変化に対する計算時間の変化が大幅に小さくなることがわかる．

方向マップの導入により明らかに小さくなることがわかる．

表 1 および図 5 の結果より，方向マップを導入した特徴抽出は，処理が高速，かつ，特徴数の変動に対する計算時間の変動が小さいという性質を持つことが明らかとなった．この性質は，高速な処理とシーン内容によらずフレームレートを一定に保つことが望ましいオンライン処理において有用と言える．実際のオンライン処理での有用性を確認するために，IEEE1394 カメラ (Point Grey Research 社 Flea2[25]) を用い，図 6 に示す時系列画像を取り込み処理を行った．入力画像サイズは QVGA サイズとし，3539 フレームに対し処理をした．図 7 に，その処理における特徴数および計算時間の変化を示す．この結果から，方向マップを導入した特徴抽出の上記の有用性が確認できたと言える．

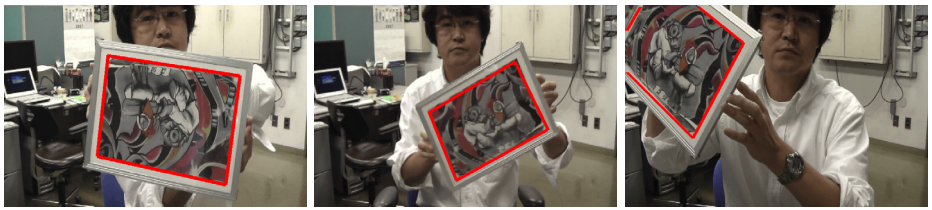


図 6 方向マップの使用の有無による計算時間の変化を計測するために用いた時系列画像の一部．画像中の赤い枠は，本論文での実装により得られた特徴量を用い，文献[26] の方法で物体位置検出を行った結果を示す．

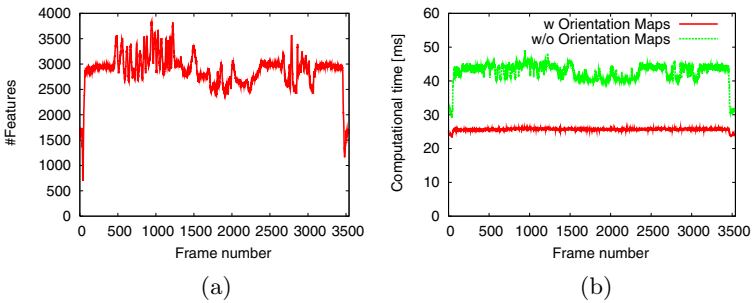


図 7 時系列画像対する実験結果．(a) 特徴数の変化，(b) 計算時間の変化．

方向マップを用いた実装により抽出された局所不変特徴量の応用の一例を，図 6 に示す．この図の赤い枠は，画像”Graffiti” を認識対象とし，文献[26] の方法でその位置検出を行った結果である．回転，スケール変化，隠れ等の見えの変化が生じる状態でも，位置検出が可能なことが確認できた．

6. むすび

本論文では，GPU と方向マップに基づく局所不変特徴量抽出の高速化について検討を行った．その結果，記述子の計算に方向マップを導入することにより，高速な処理，および，特徴数の変動に対する計算時間の変動が小さい，というオンライン処理において望ましい性質を有する特徴抽出が実現できることを示した．

数多くの特徴を高速に抽出できることは利点であるが，その反面，画像の対応付けや物体

© 2011 Information Processing Society of Japan

認識において計算量が増加する一要因となる．よって，今後は，特徴抽出と共に，対応付けや物体認識の高速化に関しても研究開発を進める予定である．

謝辞

本研究の一部は，科学研究費補助金，課題番号 22500171「並列処理に基づく物体認識アルゴリズムに関する研究」の助成の下で行われた．

参 考 文 献

1) Schmid, C. and Mohr, R.: Local greyvalue invariants for image retrieval, *IEEE Trans. PAMI*, Vol.19, No.5, pp.530–535 (1997).
2) Lowe, D.: Distinctive image features from scale-invariant keypoints, *Int. J. Comp. Vis.*, Vol.60, No.2, pp.91–110 (2004).
3) Mikolajczyk, K. and Schmid, C.: A performance evaluation of local descriptors, *IEEE Trans. PAMI*, Vol.27, No.10, pp.1615–1630 (2005).
4) Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Gool, L.V.: A comparison of affine region detectors, *Int. J. Comp. Vis.*, Vol.65, No.1/2, pp.43–72 (2005).
5) Mikolajczyk, K., Zisserman, A. and Schmid, C.: Shape recognition with edge-based features, *Proc. Britich Machine Vis. Conf.*, Vol.2, pp.779–788 (2003).
6) Csurka, C., Dance, C.R., Fan, L., Willamowski, J. and Bray, C.: Visual categorization with bags of keypoints, *Proc. Workshop on Statistical Learning in Computer Vision*, pp.1–22 (2004).
7) Dalal, N. and Triggs, B.: Histograms of orientated gradients for human detection, *Proc. Int. Conf. Comp. Vis. Patt. Recog.*, Vol.1, pp.886–893 (2005).
8) Fei-Fei, L. and Perona, P.: A Bayesian hierarchical model for learning natural scene categories, *Proc. Int. Conf. Comp. Vis. Patt. Recog.*, Vol.2, pp.524–531 (2005).
9) Ma, X. and Grimson, W.E.: Edge-based rich representation for vehicle classification, *Proc. Int. Conf. Comp. Vis.*, Vol.2, pp.1185–1192 (2005).
10) Jurie, F. and Triggs, B.: Creating efficient codebooks for visual recognition, *Proc. Int. Conf. Comp. Vis.*, Vol.1, pp.604–610 (2005).
11) Nowak, E., Jurie, F. and Triggs, B.: Sampling strategies for bag-of-features image classification, *Proc. European Conf. Comp. Vis.*, pp.490–503 (2006).
12) Mikolajczyk, K., Leibe, B. and Schiele, B.: Multiple object class detection with a generative model, *Proc. Int. Conf. Comp. Vis. Patt. Recog.*, Vol.1, pp.26–36 (2006).
13) Lampert, C.H., Blaschko, M.B. and Hofmann, T.: Efficient subwindow search: A branch and bound framework for object localization, *IEEE Trans. PAMI*, Vol.31, No.12, pp.2129–2142 (2009).
14) 市村直幸：密なエッジサンプリングに基づく局所不変特徴量による対応付け，信学技報， No.PRMU2009-51, pp.71–76 (2009).
15) 市村直幸：GPU による特徴点とエッジに基づく局所不変特徴量の抽出，情処研報，Vol.2009–CG–136, No.11 (2009).
16) Tola, E., Lepetit, V. and Fua, P.: DAISY: An efficient dense descriptor applied to wide-baseline stereo, *IEEE Trans. PAMI*, Vol.32, No.5, pp.815–830 (2010).
17) J SPORTS において放送された映像を使用している．
18) Lindeberg, T.: Feature detection with automatic scale selection, *Int. J. Comp. Vis.*, Vol.30, No.2, pp.79–116 (1998).
19) CUDA Zone: http://www.nvidia.com/object/cuda_home_new.html.
20) Ando, S.: Consistent gradient operators, *IEEE Trans. PAMI*, Vol. 22, No. 3, pp. 252–265 (2000).
21) Trajkovic, M. and Hedley, M.: Fast corner detection, *Image and Vision Computing*, Vol.16, pp.75–87 (1998).
22) 市村直幸：近似 LoG フィルタを用いた局所不変特徴量の抽出 – GPU による実装–，情処研報， Vol.2008-CVIM-165, pp.243–250 (2008).
23) *NVIDIA CUDA Programming Guide, Version 3.2, p.14* (2010).
24) Feature detector evaluation sequences: http://lear.inrialpes.fr/people/mikolajczyk/Database/.
25) Point Grey Research, Inc.: http://www.ptgrey.com/index.asp.
26) Ichimura, N.: Recognizing multiple billboard advertisements in videos, *Proc. Pacific-Rim Symp. on Image and Video Technology (PSIVT)*, pp.463–473 (2006).

付　　録

A.1 方向マップに適用するガウシアンフィルタのスケールの決定方法

図 2 に示す 4×4 のセルを局所領域に配置する場合について，方向マップに適用するガウシアンフィルタのスケールの決定方法を示す．局所領域の固有スケールを σ とする．この σ にスケール係数 s を乗じて，局所領域の大きさを決定する．局所領域を 4×4 に分割すると，その結果得られる正方形の幅は $s\sigma/4$ となる．

方向マップに対するガウシアンフィルタのスケールを σ_o とする．局所領域を分割して得られる正方形の外接円の位置で，ガウス関数がどの程度の値を保持するかを変化させ，セル内部の輝度勾配の大きさに対する重み付けを制御する．本論文では，外接円の位置でのガウス関数の定義域の値を $3\sigma_o$ とした．よって，外接円の直径は $6\sigma_o$ となる．この直径が正方形の幅の $\sqrt{2}$ 倍と一致することから，σ_o は次式で求められる．

$$\sigma_o = \sqrt{2}s\sigma/24 \quad (1)$$

表 1 の計算時間の計測では $s = 20$ とし，$\sigma_o \approx 1.2\sigma$ より定まるスケールを有するガウシアンフィルタを多重解像度方向マップに適用した．

© 2011 Information Processing Society of Japan