

Space-Time Patch を用いた物体の移動方向識別とセグメンテーション

村井泰裕[†] 藤吉弘巨[†] 金出武雄^{††}[†] 中部大学工学研究科 〒 487-8501 愛知県春日井市松本町 1200^{††} カーネギーメロン大学ロボティクス工学研究所 〒 15213-3890 ペンシルバニア州ピッツバーグ市

E-mail: †yasu@vision.cs.chubu.ac.jp, hf@cs.chubu.ac.jp, ††tk@cs.cmu.edu

あらまし 本稿では, space-time patch 特徴を用いて物体の移動方向識別とセグメンテーションを同時に行う手法を提案する. 提案手法は, 複数の学習サンプルを用いて作成した space-time patch 特徴のコードブックを参照してベクトル量子化することで, 移動方向識別と対象物体の重心位置推定を行う. そして, 推定された物体の重心位置から, 学習サンプルのマスク画像を用いた確率計算に基づいて物体領域をセグメンテーションを行う. これにより, 移動方向が異なる物体が交差する際にオクルージョンが発生していても, 移動方向を識別しているためセグメンテーションが可能となる. 評価実験の結果, アピランス情報を用いた従来法より, オクルージョン発生時における対象物体の検出率を向上させることができた.

キーワード Space-time patch, 物体検出, 移動方向識別, 領域セグメンテーション

Classifying Direction of Movement and Segmenting Regions of Objects by Using Space-Time Patches

Yasuhiro MURAI[†], Hironobu FUJIYOSHI[†], and Takeo KANADE^{††}[†] Dept. of Computer Science, Chubu University. Matsumoto 1200, Kasugai, Aichi, 487-8501 Japan^{††} The Robotics Institute, Carnegie Mellon University. Pittsburgh, Pennsylvania, 15213-3890 USA

E-mail: †yasu@vision.cs.chubu.ac.jp, hf@cs.chubu.ac.jp, ††tk@cs.cmu.edu

Abstract We developed a method for classifying the direction of movement and for segmenting objects simultaneously using features of space-time patches. Our approach uses vector quantization to classify the direction of movement of an object and to estimate its centroid by referring to a codebook of the space-time patch feature, which is generated from multiple learning samples. We segmented the objects' regions based on the probability calculated from the mask images of the learning samples by using the estimated centroid of the object. Even though occlusions occur when multiple objects overlap in different directions of movement, our method detects objects individually because their direction of movement is classified. Experimental results show that object detection is more accurate with our method than with the conventional method, which is only based on appearance features.

Key words Space-time patch, Object detection, Classifying direction of movement, Regions segmentation

1. はじめに

近年, 監視映像からの不審者の発見や, ITS における歩行者の安全のため, カメラで捉えた画像から自動的に対象物体を識別・領域分割を行う技術の実現が期待されている. Fujiyoshi 等は, 屋外に設置したカメラ映像からフレーム間差分により検出された移動物体の形状から, ニューラルネットワークを用いて人や車に識別し, その結果を地図上へマッピングすることにより, 歩行者であるユーザの移動支援を行う RCT システム (Robotic Communication Terminals) [1] を提案している. ま

た, 藤本等は, 店舗内の全方位視覚センサから得た映像から顧客の位置や移動方向を推定し, その情報を用いて空間マップを作成することにより, 顧客全体の店内における分布, 各顧客の移動軌跡および停留状況などをユーザに提示するシステム [2] を提案している. この様な画像処理技術を用いたシステムでは, 画像上の移動物体の動きの情報を得ることや, 対象物体の領域を検出することは非常に重要な問題である.

画像上の移動物体の検出・動きの解析を行う手法として, 物体の動きの移動量をベクトルデータとして表現したオプティカルフロー [3] が挙げられる. しかし, オプティカルフローでは

ノイズが発生しやすいことや、非剛体の物体に対する動きの解析やテクスチャの異なる物体同士の動きの評価が困難である等の問題がある。これらの問題に対して Shechtman 等は、テンプレート映像と入力映像の時空間画像から抽出された 2 つの space-time patch (ST-patch) の相関関係を複数用いることにより、テンプレート映像の物体と同じ動きをしている物体を入力映像から識別する手法 [4] を提案している。この手法で用いられている ST-patch の相関関係は、テクスチャが異なる対象物体の動きや、非剛体の物体の様に複雑に動く物体に対しても動きの評価を行うことができる。また、このような局所的な時系列情報の学習を行うことにより物体の動きの識別を行う手法 [5] も提案されている。

対象物体の検出法には、物体の局所的な特徴を用いた part-based approach がある。Agarwal 等は学習サンプル画像の特徴点の周辺領域のパッチを抽出し、類似したアピランス情報を持つパッチのクラス分けをしたコードブックを作成し、入力画像の特徴点の周辺領域のパッチとコードブックをマッチングさせることにより物体検出を行う手法 [6] を提案している。Leibe 等は各パッチに物体の重心位置、マスク画像を格納したコードブックの作成を行い、コードブック内の各学習サンプルパッチに格納してある物体の重心位置、マスク画像から物体の識別とセグメンテーションを行う手法 [7] [8] を提案している。Opelt 等は物体の輪郭の学習サンプルから物体の重心位置の投票により重心位置推定を行い、重心位置に投票された物体の輪郭の位置関係から AdaBoost によって識別することで、物体検出を行う手法 [9] を提案している。これらの手法は、物体の部分的な特徴から学習サンプルを用いて対象物体の検出を行うことで、オクルージョンへの対応が可能となるメリットがある。また、局所領域における輝度の勾配方向をヒストグラム化した特徴ベクトルである HOG (Histograms of Oriented Gradients) を用いた人検出法 [10] や、矩形領域の勾配を表す Haar-like 特徴を用いた顔検出法 [11] 等、part-based approach を用いた手法は数多く提案されている。

このような手法では、主に 1 枚の画像のアピランス情報に基づいて対象物体の検出を行っている。本研究では、局所的なアピランス (見え) とモーション (動き) の情報を持つ ST-patch 特徴を用いて、物体の移動方向識別とセグメンテーションを同時に行う手法を提案する。提案手法では、まず複数の学習サンプルより作成した ST-patch 特徴のコードブックを参照することにより、入力画像から得られる ST-patch 特徴をベクトル量子化し、移動方向識別を行う。次に、学習サンプル ST-patch 特徴に格納してある物体の重心位置を、ST-patch 特徴の移動方向識別結果に応じて別々に投票していくことで物体の重心位置推定を行う。そして、推定された物体の重心位置から、学習サンプル ST-patch 特徴に格納してある物体のマスク画像を用いることで物体のセグメンテーションを実現する。

2. ST-patch

本研究では Shechtman 等の提案手法で用いられている ST-patch 特徴 [4] を用いて移動方向識別を行う。ST-patch 特徴は、

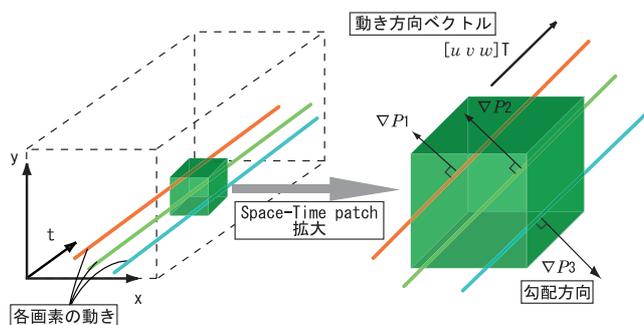


図 1 ST-patch の概要

ST-patch 内の画素の動きが異なる場合 (例えば、右移動と左移動)、それぞれ異なる特徴が得られる。故に、複数の学習サンプルより ST-patch 特徴を抽出し、類似した ST-patch 特徴をクラスタリングすることにより、動きが異なる ST-patch 特徴のクラスタで分けられたコードブックを作成することができる。本章では、移動方向識別ために用いる ST-patch 特徴について、ST-patch 特徴の性質について、学習サンプルより抽出した ST-patch 特徴のコードブックの作成方法について述べる。

2.1 ST-patch の概要

ST-patch 特徴は、画像を時間方向に重ねた 3 次元データである時空間画像の局所領域から得られる特徴である。図 1 に ST-patch の概念を示す。x, y は画像の座標軸, t は時間軸, 3 つのラインは個々の画素の動き, $[u \ v \ w]^T$ は ST-patch 中の動き方向ベクトル, ∇P_i は個々の画素の勾配方向ベクトルを表している。

2.2 ST-patch 特徴

ST-patch 特徴を得るため、時空間画像においての x 軸, y 軸, t 軸の勾配を求める。画像中の動きが一定の場合、各軸に対するある画素 i の勾配 $\nabla P_i = (P_{x_i}, P_{y_i}, P_{t_i})$ は、画素の動き方向ベクトル $[u \ v \ w]^T$ に対して垂直となる。よって、式 (1) の関係が成り立つ。

$$\nabla P_i \begin{bmatrix} u \\ v \\ w \end{bmatrix} = 0 \quad (1)$$

画素数が n の場合、式 (1) は式 (2) となる。

$$\begin{bmatrix} P_{x_1} & P_{y_1} & P_{t_1} \\ P_{x_2} & P_{y_2} & P_{t_2} \\ \vdots & \vdots & \vdots \\ P_{x_n} & P_{y_n} & P_{t_n} \end{bmatrix}_{n \times 3} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n \times 1} \quad (2)$$

ST-patch 中の n 画素の ∇P_i からなる $n \times 3$ の行列を \mathbf{G} とし、行列 \mathbf{G}^T を掛けると式 (3) となる。

$$\mathbf{G}^T \mathbf{G} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}_{3 \times 1} \quad (3)$$

この様に、行列 $\mathbf{G}^T \mathbf{G}$ は 3×3 の行列となる。ここで、行列 $\mathbf{G}^T \mathbf{G}$ を \mathbf{M} とすると式 (4) のように表すことができる。

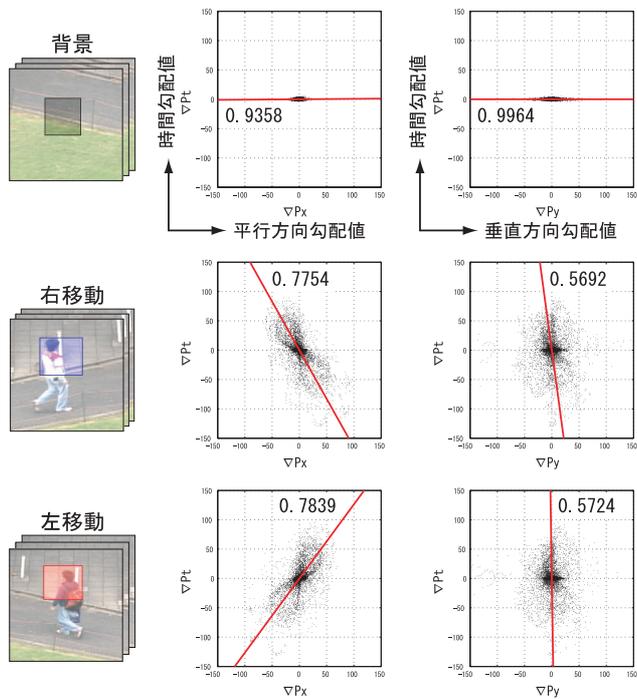


図 2 ST-patch 特徴の性質

$$M = G^T G = \begin{bmatrix} \sum P_x^2 & \sum P_x P_y & \sum P_x P_t \\ \sum P_y P_x & \sum P_y^2 & \sum P_y P_t \\ \sum P_t P_x & \sum P_t P_y & \sum P_t^2 \end{bmatrix} \quad (4)$$

式 (4) より求められる行列 M が 1 つの ST-patch から得られる特徴となる。行列 $M(3 \times 3)$ には、左上の 2×2 の行列にアピランスの情報、3 行目と 3 列目にはモーションの情報を持つ。この行列 M を 9 次元ベクトルとして表現した行列 m を ST-patch 特徴とする。

$$m = \left(\sum P_x^2, \sum P_x P_y, \dots, \sum P_t^2 \right) \quad (5)$$

2.3 ST-patch 特徴の性質

図 2 に背景、右移動、左移動している物体のある局所領域の x 軸、 y 軸、 t 軸の勾配値である $\nabla P_i = (P_{x_i}, P_{y_i}, P_{t_i})$ をそれぞれプロットしたグラフを示す。それぞれの画像中の矩形領域は勾配値を求めた ST-patch の領域 ($50[\text{pixel}] \times 50[\text{pixel}] \times 3[\text{frame}]$)、グラフの赤線と数値は主成分分析により求められた第 1 主成分の軸とその寄与率を表している。第 1 主成分の軸はプロットされた勾配値の分布の傾向を示し、寄与率が高いほどその軸方向への分布が強いことを意味する。背景のグラフを見てみると、横方向に分布の傾向を表す軸が出ており、寄与率も高いことが分かる。これは、動きの無い背景では時間勾配値が極めて低いためである。右移動と左移動の垂直方向勾配値と時間勾配値のグラフでは、縦方向へ分布の傾向を表す軸が出てくるが、寄与率は低い値となっている。これより、垂直方向勾配値と時間勾配値の分布は広く分散していると言える。次に、右移動と左移動の水平方向勾配値と時間勾配値の分布のグラフを見ると、それぞれの分布の傾向を表す軸が対角方向になってい

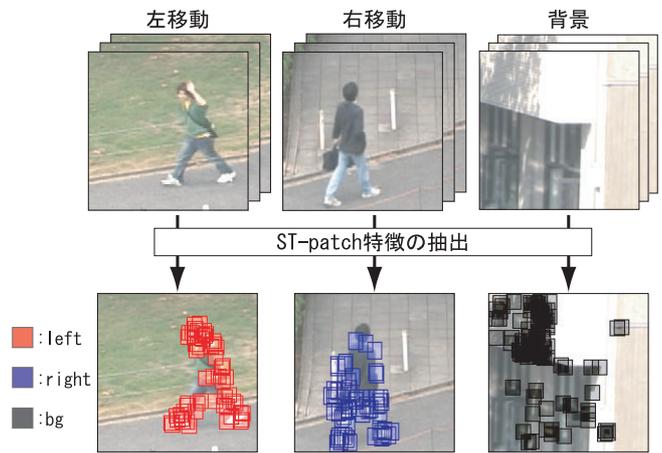


図 3 学習サンプル ST-patch 特徴の抽出例

ることが分かる。また、その寄与率は共に高い値となっており、それぞれの軸方向へ強く分布していることが分かる。この様に、移動方向の異なる物体の勾配値からはそれぞれ違った特徴を得ることができる。このような性質を 9 次元ベクトルとして表した特徴量が ST-patch 特徴である。

2.4 ST-patch 特徴のコードブック

ST-patch より得られる 9 次元ベクトル m のコードブックを作成するために、本研究では LBG アルゴリズム [12] を用いる。これにより、学習サンプルの特徴ベクトルを N 個の代表ベクトルの組にクラスタリングすることができる。ST-patch 特徴のコードブック作成には、複数枚の画像を学習サンプルとして用いる。図 3 に学習サンプル ST-patch 特徴の抽出例を示す。また、この様に抽出された学習サンプル ST-patch 特徴を用いて ST-patch 特徴のコードブックを作成する流れを以下に述べる。

- Step1 複数の学習サンプルから ST-patch 特徴を抽出
- Step2 抽出された ST-patch 特徴に移動方向ラベル、物体の重心位置、物体のマスク画像を格納
- Step3 LBG アルゴリズムにより N クラスタに分けたコードブックを作成
- Step4 コードブックのクラスタのラベル含有率を計算

学習サンプル ST-patch 特徴は、予め作成した物体のマスク画像の領域内の ST-patch 特徴を移動物体の学習サンプルとして抽出する。そして、学習サンプル ST-patch 特徴を抽出した後、正解の移動方向ラベル、マスク画像より求められる物体の重心位置、物体のマスク画像をそれぞれ格納しておく。これらの情報は、物体の重心位置推定、物体のセグメンテーションに用いる。これらの学習サンプル ST-patch 特徴を、LBG アルゴリズムを用いてコードブックを作成する。しかし、作成されたコードブックのクラスタに属した学習サンプルの各移動方向ラベルは、全て同じラベルになるとは限らない。これは、LBG アルゴリズムではユークリッド距離の近い学習サンプル ST-patch 特徴のクラスタに分けているためである。そのため、図 4 に示すように、各クラスタに属する移動方向ラベルの割合を求める。本研究では、これらをクラスタのラベル含有率と定義する。こ

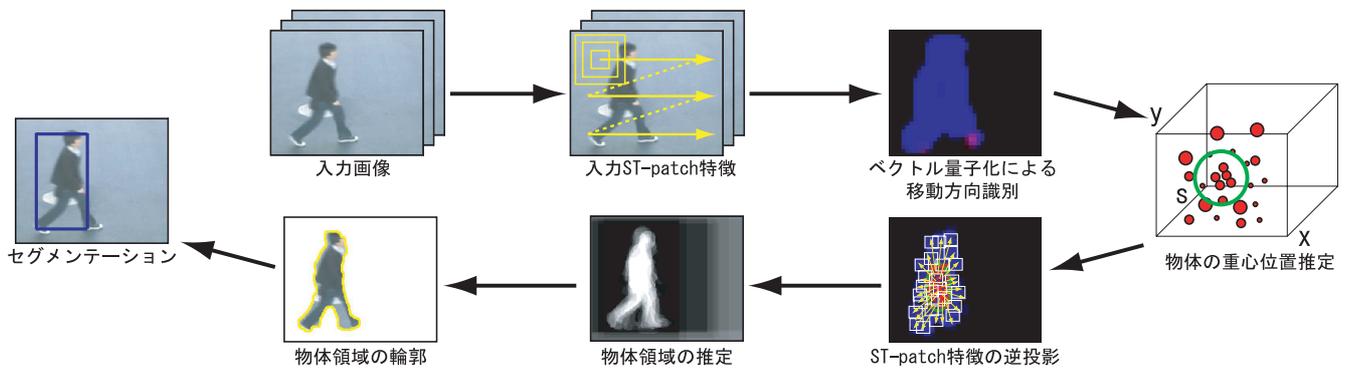


図 5 物体の移動方向識別とセグメンテーションの流れ

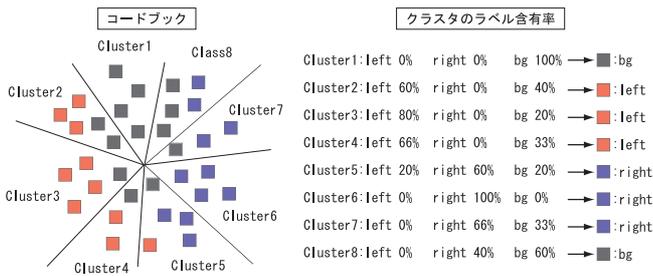


図 4 ラベル含有率の概要

のコードブックのクラスタのラベル含有率に基づいて移動方向の識別を行う。

3. 物体の移動方向識別とセグメンテーション

本研究では、学習サンプルより作成した ST-patch 特徴のコードブックを用いて、入力画像から得られる ST-patch 特徴をベクトル量子化し、移動方向識別を行う。そして、Leibe 等によって提案されている手法 [7] を応用し、識別された移動方向の結果と学習サンプル ST-patch 特徴から物体の重心位置の Voting 処理を行い、そこから重心位置の推定をする。推定された物体の重心位置周辺に投票された学習サンプル ST-patch 特徴に対応したマスク画像を用いて物体のセグメンテーションを行う。図 5 に物体の移動方向識別とセグメンテーションの流れを示す。本章では、物体の移動方向識別とセグメンテーションを行うための各処理の詳細について述べる。

3.1 ST-patch 特徴のベクトル量子化

入力画像より抽出された ST-patch 特徴を、作成したコードブックを用いてベクトル量子化する。入力 ST-patch 特徴のベクトル量子化による移動方向識別の流れを以下に述べる。

- Step1 入力画像のラスタスキャンにより抽出されたパッチを学習サンプルと同じサイズにダウンサンプリングし、そのパッチの ST-patch 特徴を抽出
- Step2 入力 ST-patch 特徴を e 、作成したコードブックの各クラスタの特徴ベクトルを c とし、2 つ特徴ベクトル間のユークリッド距離を計算し、ユークリッド距離が最小のクラスタ C を式 (6) により算出

$$C = \underset{c}{\operatorname{argmin}} \| e - c \|^2 \quad (6)$$



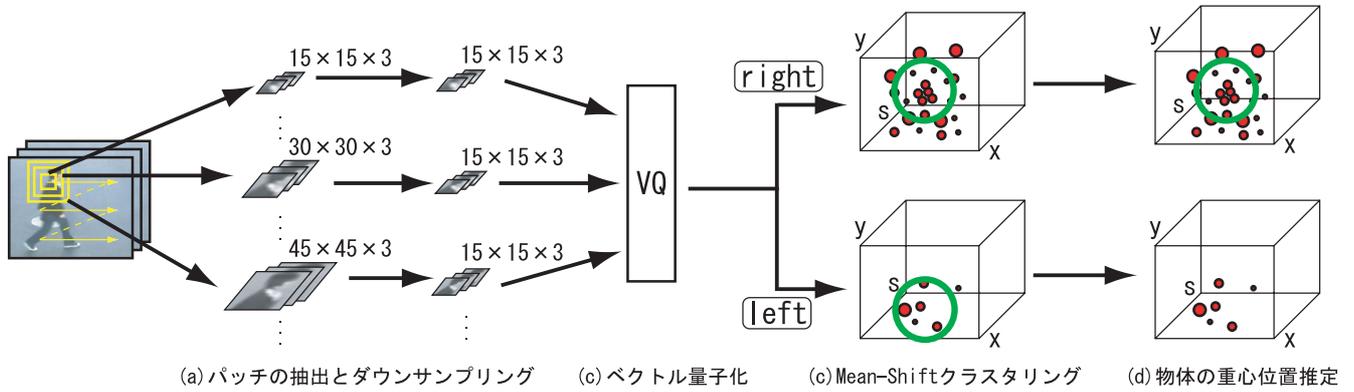
図 6 移動方向識別された領域の抽出例

- Step3 コードブックのクラスタのラベル含有率に基づいて移動方向の識別
- Step4 入力画像から抽出するパッチのサイズを変更
- Step5 画像全体の処理が終了するまで Step1 ~ Step4 を繰り返す

この様にパッチのサイズの変更し、繰り返して処理を行うことにより、物体のスケール変化に対応させることができる。図 6 に入力 ST-patch 特徴のベクトル量子化による移動方向識別結果と、移動方向識別された領域をそれぞれ抽出した例を示す。この様に、ST-patch 特徴を用いることで移動方向の識別を行うことが可能である。

3.2 物体の重心位置推定

作成したコードブックを用いて入力 ST-patch 特徴をベクトル量子化すると、入力 ST-patch 特徴のコードブックのクラスタのラベル含有率を基にして移動方向が求められるが、物体全体の移動方向の識別を行うことはできない。また、入力 ST-patch 特徴をベクトル量子化した際にアウトライヤを含む場合がある。そこで本研究では、物体の移動方向識別を行うため、ベクトル量子化により求められたコードブックのクラスタに属する学習サンプル ST-patch 特徴に格納されている物体の重心位置を投票し、Mean-Shift クラスタリング [13] により投票点の極値を求める。そして、NN 法 (Nearest Neighbor Method) により閾



(a)パッチの抽出とダウンサンプリング (b)ベクトル量子化 (c)Mean-Shiftクラスタリング (d)物体の重心位置推定
 図7 物体の重心位置推定

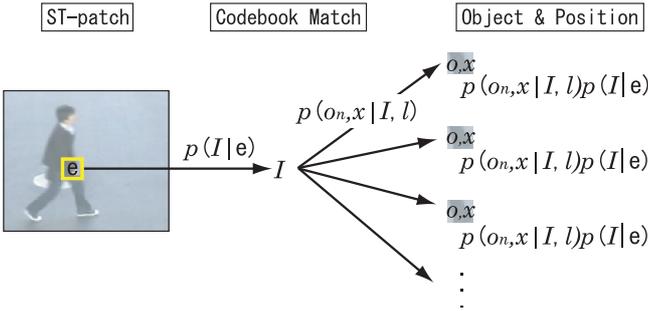


図8 重心位置の投票の概要

値以内の極値を統合することで物体の重心位置推定を行う。

3.2.1 重心位置の投票

物体の重心位置を推定するために Voting 処理を行う。図8に重心位置の投票の概要を示す。画像座標 l より抽出された入力 ST-patch 特徴 e のベクトル量子化を行うとクラス I が得られる。そして、求められたクラス I から、物体 o_n の重心位置 x である確率 $p(o_n, x | I, l)$ を得ることができる。よって、ある物体 o_n ($n =$ 右移動の物体, 左移動の物体, ...) の重心位置 x である確率を式 (7) より求める。

$$p(o_n, x | e, l) = p(o_n, x | e, I, l) p(I | e, l) \quad (7)$$

ここで、 $p(I | e, l)$ の確率は、入力 ST-patch 特徴をベクトル量子化して求められたコードブックのクラス I のラベル含有率を $p(I | e, l)$ の確率として用いる。そして、複数の ST-patch (k 個) から得られるウィンドウ $W(x)$ 内の投票点の確率のスコアは式 (8) により得られる。

$$score(o_n, x) = \sum_k \sum_{x_j \in W(x)} p(o_n, x_j | e_k, l_k) \quad (8)$$

3.2.2 重心位置の推定

図7に重心位置推定の流れを示す。入力画像より抽出されたパッチをダウンサンプリングし、そのパッチから抽出された ST-patch 特徴 (図7(a)) のベクトル量子化によって移動方向識別 (図7(b)) を行い、3.2.1 で示した Voting 処理をする。そして、識別された移動方向ごとに投票点が最も密な位置を3次元 Mean-Shift クラスタリング ($x, y, scale$) により探索することで物体の重心位置を推定する。Mean-Shift クラスタリングに

より収束した極値 (図7(c)) を、NN 法により一定の閾値でクラスタリングを行い、一つのクラスに統合する。NN 法により統合された極値周辺の投票点の重みの合計が閾値以下である極値をリジェクトすることにより、アウトライアの除去を行う (図7(d))。この様に求められた位置を物体の重心位置とする。

3.2.3 Mean-Shift クラスタリング

注目点を x 、各投票点の位置を x_i ($i = 1, \dots, n$)、ウィンドウの半径を h 、カーネル関数を k とすると、Mean-Shift ベクトル $m_h(x)$ は式 (9) により求められる。

$$m_h(x) = \frac{\sum_{i=1}^n x_i k\left(\left\|\frac{x-x_i}{h}\right\|^2\right)}{\sum_{i=1}^n k\left(\left\|\frac{x-x_i}{h}\right\|^2\right)} - x \quad (9)$$

カーネル関数 k は式 (10) のように定義する。

$$k(x) = \exp\left(-\frac{1}{2} \|x\|^2\right) \quad (10)$$

この様なカーネル関数 k を用いることにより、注目点と投票された点が近いほど大きな値を返し、投票された点が遠いほど小さな値を返すことができ、投票された点の極値を探索することができる。そして、ある時点で求められた Mean-Shift ベクトル $m_h(x^t)$ を用いて、注目点を式 (11) により移動させる。

$$x^{t+1} = x^t + m_h(x^t) \quad (11)$$

求められた Mean-Shift ベクトル $m_h(x^t)$ が閾値以下に収束するまで繰り返すことにより、物体の重心位置推定を行う。

3.3 物体領域のセグメンテーション

図9に物体領域のセグメンテーションの流れを示す。この様に、推定された物体の重心位置周辺に投票した学習サンプル ST-patch 特徴と、そこに格納されている物体のマスク画像を用いて物体領域を推定し、物体のセグメンテーションを行う。

3.3.1 ST-patch 特徴の逆投影

推定した物体の重心位置周辺へ投票した ST-patch 特徴を逆投影を行う。これにより、重心位置外の投票点 (アウトライア) の除去を行い、信頼性の高い情報を選択することができる。逆投影された学習サンプル ST-patch 特徴の各投票点が、ある物体 o_n の重心位置 x である確率 $p(o_n, x | e, l)$ の事後確率を式 (12) により求める。

$$p(e, l | o_n, x) = \frac{p(o_n, x | e, l) p(e, l)}{p(o_n, x)} \quad (12)$$

ここで、 $p(o_n, x | e)$ は式 (7) より求められる。

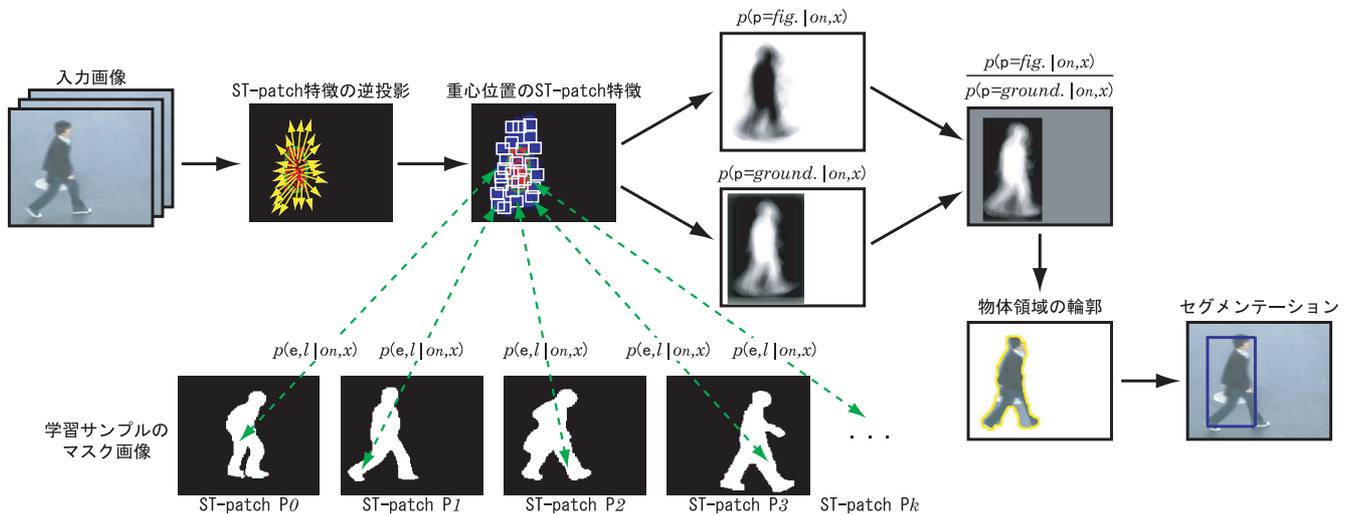


図 9 物体領域のセグメンテーション

3.3.2 物体領域の推定

物体領域の推定を行うために、まずピクセル p が物体である確率 $p(p = fig. | o_n, x)$ を求める。各ピクセルが物体である確率は式 (13) で定義するように、各ピクセルにおいて物体、背景である情報と、式 (12) で定義した事後確率を掛けることにより求めることができる。

$$p(\mathbf{p} = fig. | o_n, x) = \sum_{\mathbf{p}} p(\mathbf{p} = fig. | o_n, x, \mathbf{e}, l) p(\mathbf{e}, l | o_n, x) \quad (13)$$

ここで、各ピクセルが物体、背景である情報 $p(\mathbf{p} = fig. | o_n, x, \mathbf{e}, l)$ は、各クラスに属する学習サンプル ST-patch 特徴に格納されているマスク画像を用いることにより求められる。よって、各ピクセルが物体である確率は式 (14) により求めることができる。

$$p(\mathbf{p} = fig. | o_n, x) = \sum_{\mathbf{p}} p(\mathbf{p} = fig. | o_n, x, \mathbf{e}, l) p(\mathbf{e}, l | o_n, x) = \sum_{\mathbf{p}} p(\mathbf{p} = fig. | o_n, x, l) \frac{p(o_n, x | l) p(l | \mathbf{e}) p(\mathbf{e}, l)}{p(o_n, x)} \quad (14)$$

同様に、各ピクセルにおいて背景である確率 $p(\mathbf{p} = ground. | o_n, x)$ を式 (15) により求める。

$$p(\mathbf{p} = ground. | o_n, x) = \sum_{\mathbf{p}} (1 - p(\mathbf{p} = fig. | o_n, x, l)) p(\mathbf{e}, l | o_n, x) \quad (15)$$

式 (14), (15) では、各ピクセルにおいて物体、背景である確率の加重平均が求められる。次に、各ピクセルにおいて物体である確率と背景である確率から式 (16) により L を求める。

$$L = \frac{p(\mathbf{p} = fig. | o_n, x)}{p(\mathbf{p} = ground. | o_n, x)} \quad (16)$$

この様に求められた各ピクセルの物体である確率と背景である確率の関係を用いて、物体のセグメンテーションを行う。式

表 1 学習サンプル ST-patch 特徴の数

	シーケンス	
	歩行者	車両
背景	36,982	20,047
右移動	10,198	9,885
左移動	10,220	9,968

(16) により求められた各ピクセルの物体である確率と背景である確率の関係が閾値以下のピクセルを背景、閾値以上のピクセルを物体と表現することにより、物体領域を矩形によりセグメンテーションをすることが可能となる。

4. 評価実験

提案手法を用いて物体の移動方向識別とセグメンテーションの実験を行う。評価実験では提案手法の有効性を示すため、アピランス情報を用いて物体の識別とセグメンテーションを行う従来法 [7] との比較を行う。また、提案手法の処理時間の計測結果を示す。

4.1 実験概要

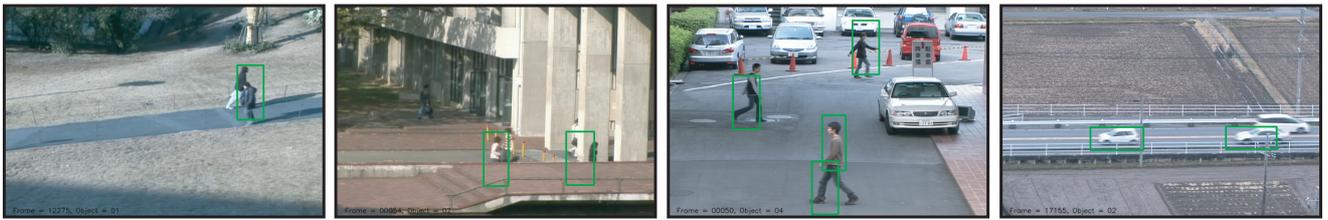
表 1 に歩行者、車両それぞれの学習サンプルから抽出した ST-patch 特徴の数を示す。このような数の学習サンプルを用いてそれぞれのコードブックを作成し、物体の移動方向識別とセグメンテーションを行う。実験では、ST-patch のサイズを $15[\text{pixel}] \times 15[\text{pixel}] \times 3[\text{frame}]$ とし、LBG アルゴリズムにより作成する ST-patch 特徴のコードブックサイズを 512 とする。

従来法と提案手法の比較実験に用いるシーケンスには、固定 DV カメラにより学習サンプルと異なる場所で撮影された、歩行者が左右に移動している映像、車両が左右に移動する 2 車線の道路の映像を用いる。

提案手法の処理時間の計測には CPU: Intel Xeon 3.6GHz, メモリ: 2.0GB を搭載した PC を使用する。また、 720×480 の評価画像 100 フレーム分の処理時間の平均から提案手法の処理時間を求める。

従来法 [7]

□: object

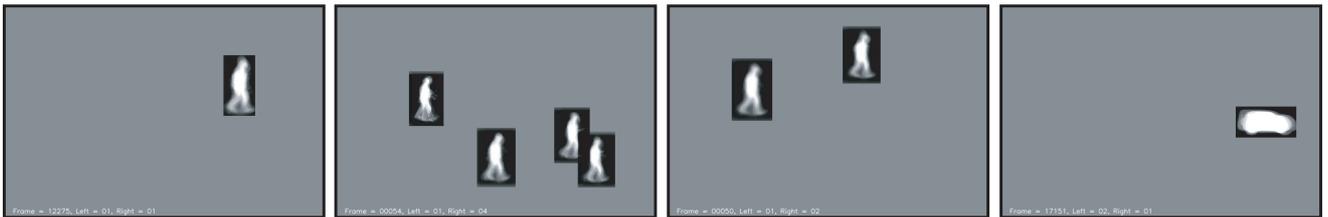


提案手法

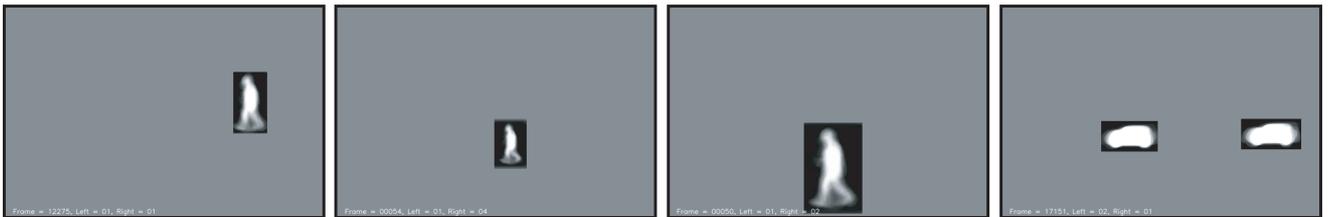
□: left □: right



右移動の物体領域推定



左移動の物体領域推定



(a)

(b)

(c)

(d)

図 10 物体の移動方向識別とセグメンテーション

4.2 実験結果

図 10 に従来法による物体の検出とセグメンテーション結果と、提案手法による物体の移動方向識別とセグメンテーション、右移動の物体領域推定、左移動の物体領域推定の結果を示す。結果より、提案手法を用いることで、物体の移動方向の識別と物体のセグメンテーションを同時に行うことができる。また、図 10(a) に示すように、異なる方向へ移動している物体が重なる場合においても、個々に物体領域のセグメンテーションを行うことができる。これは、移動方向識別結果を基に物体の重心位置の推定を別々に行い、物体のセグメンテーションをしているためである。また、提案手法は影の影響を受けにくいことが分かる。図 10(b) では、下半身にオクルージョンがある歩行者や、柱に体の一部が隠れている歩行者においても、物体領域を考慮して物体のセグメンテーションを行うことができています。これは、画像上に写っている歩行者の部分的な情報を基に物体の重心位置も求め、そこから物体のセグメンテーションをしているためである。図 10(c) は、物体のスケールが変化した場合の結果である。この様に、提案手法では物体のスケール変化に対応した物体の移動方向の識別と物体のセグメンテーションを行うことができる。図 10(d) では、学習サンプルとして車両を

用いた場合の結果である。提案手法では、学習サンプルをそれぞれ用意し、そのコードブックを作成することで、様々な物体の移動方向の識別と物体のセグメンテーションを行うことが可能である。

表 2 に提案手法と従来法を用いて物体の検出を行った比較実験結果を示す。比較実験には、歩行者に対する比較として図 10(a) の場所で撮影したシーケンス、車両に対する比較として図 10(d) の場所で撮影したシーケンスを用いた。識別対象フレームは画像中に物体が存在しているフレームのみとし、ある程度物体領域のセグメンテーションを行うことができていないフレームを正解とする。結果より、歩行者と比べ、車両の検出率が高いことが分かる。これは、歩行者は複数人が重なった状態で同一方向へ移動するケースが多いが、車両は重なった状態で同一方向へ移動するケースが無いためである。そして、提案手法はアピランス情報のみを用いる従来法と比べ、歩行者、車両ともに検出率が向上していることが分かる。従来法では物体が重なった状態の検出を行うことは困難であるが、提案手法では移動方向が異なる物体が重なった状態においても検出することができるため、従来法より検出率を向上させることができた。

表 3 に提案手法の処理時間を示す。ここで、スケール対応有

表 2 検出結果 [frame]

	シーケンス	
	歩行者 (10,134)	車両 (12,963)
従来法 [7]	64.3% (6,519)	70.7% (9,124)
提案手法	74.7% (7,567)	93.3% (12,090)

表 3 提案手法の処理時間 [sec]

	スケール対応無し	スケール対応有り
ベクトル量子化と重心位置の投票	0.48	6.02
重心位置の推定	0.03	0.03
物体のセグメンテーション	0.93	0.70
合計	1.44	6.75

りの時のパッチサイズの変更させる間隔は 0.1 とし、学習サンプルのサイズに対して 1.0~2.5 のスケールサイズまで処理を行った。計測時間を見ると、スケール対応有りの時の ST-patch 特徴のベクトル量子化と重心位置の投票に処理時間が掛かっている。これは、物体のスケール変化へ対応させるため、画像上の同じ位置に対してパッチのサイズの変更を繰り返して処理を行っているためである。

提案手法の問題として、図 11(a) の様に同一方向へ移動している物体が重なった場合、投票された点からそれぞれの物体の重心位置の推定を行うことが困難であるため、物体のセグメンテーションに失敗することがある。これは、ST-patch 特徴の 9 次元ベクトルにおけるアピランス情報が少ないため、個々の物体の重心位置投票の精度が低く、投票点が分散してしまうためであると考えられる。そして、図 11(b) のように、学習サンプルに存在しない移動物体 (バス等) の場合、物体の識別を行うことができないため、誤識別を起こすことがある。また、推定されたスケールの精度が低いという問題もある。これは、9 次元ベクトルで表される 1 つの ST-patch 特徴の情報が少ないため、入力画像から抽出したパッチのサイズを変更していき、学習サンプルと同じサイズにダウンダウニングするだけでは、物体のスケールを表現しにくい場合があるためだと考えられる。その他、提案手法ではスケール対応有りの時の処理に時間がかかるという問題がある。

5. おわりに

本稿では、ST-patch 特徴を用いて物体の移動方向識別とセグメンテーションを同時に行う手法を提案した。提案手法では、物体にオクルージョンがある場合においても、物体領域を考慮したセグメンテーションが可能である。この様に、移動方向識別結果を基に物体のセグメンテーションを行うことで、移動方向が異なる物体が重なった状態においても物体のセグメンテーションが可能となり、アピランス情報を用いた従来法より、オクルージョン発生時における対象物体の検出率を向上させることができた。

今後は、提案手法に用いた ST-patch 特徴にアピランス情報を付加することで、より正確な物体の重心位置の投票を行い、物体のセグメンテーション精度を向上させる予定である。

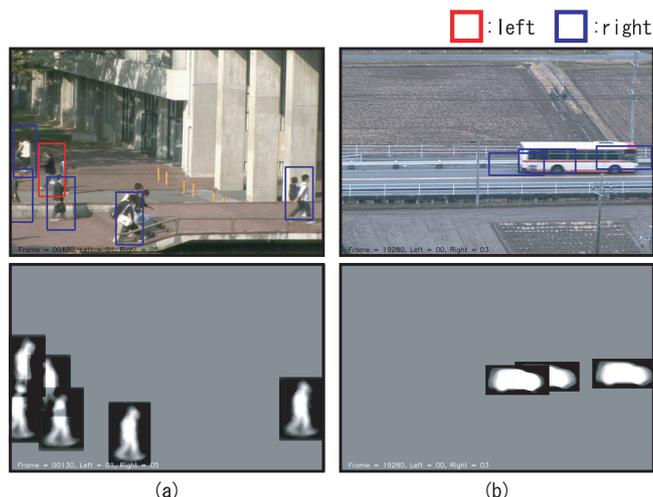


図 11 移動方向識別とセグメンテーションの失敗例

文 献

- [1] H. Fujiyoshi, T. Komura, I. E. Yairi, and K. Kayama, "Road Observation and Information Providing System for Supporting Mobility of Pedestrian", IEEE International Conference on Computer Vision Systems, pp. 37-44, 2006.
- [2] 藤本喜彦, 小原ゆう, 柴田史久, 馬場口登, 八木康史, "店舗内に設置した全方位視覚センサによる顧客の行動解析", 情報処理学会研究報告, no.142, pp.17-22, 2004.
- [3] BKP. Horn and BG. Schunck, "Determining optical flow", Artificial Intelligence, vol. 17, pp. 185-203, 1981.
- [4] E. Shechtman and M. Irani, "Space-Time Behavior Based Correlation", Computer Vision and Pattern Recognition, vol.1, pp. 405-412, 2005.
- [5] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words", British Machine Vision Conference, vol. 3, pp. 1249-1258, Sept 2006.
- [6] S. Agarwal and D. Roth, "Learning a Sparse Representation for Object Detection", European Conference on Computer Vision, pp. 113-130, 2002.
- [7] B. Leibe, A. Leonardis, and B. Schiele, "Interleaved Object Categorization and Segmentation", British Machine Vision Conference, Norwich, pp. 759-768, 2003.
- [8] B. Leibe, A. Leonardis, and B. Schiele, "Combined Object Categorization and Segmentation with an Implicit Shape Model", European Conference on Computer Vision, Prague, pp. 496-510, 2004.
- [9] A. Opelt, A. Pinz, and A. Zisserman, "Incremental learning of object detectors using a visual shape alphabet", Computer Vision and Pattern Recognition, vol.1, pp. 3-10, 2006.
- [10] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection", IEEE Computer Vision and Pattern Recognition, pp. 886-893, 2005.
- [11] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", Computer Vision and Pattern Recognition, vol.1, pp. 511-519, 2001.
- [12] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design", IEEE Trans. on Communications, vol.28, no.1, pp. 84-95, 1980.
- [13] D. Comaniciu and P. Meer, "Mean Shift Analysis and Applications", International Conference on Computer Vision, vol.2, pp. 1197-1203, 1999.