

人検出のための Real AdaBoost に基づく HOG 特徴量の効率的な削減法

松島 千佳^{†1} 山内 悠嗣^{†1}
山下 隆義^{†1,†2} 藤吉 弘 亘^{†1}

本稿では、人検出のための Real AdaBoost に基づく HOG 特徴量の効率的な削減法を提案する。提案手法は、人検出において用いられる HOG 特徴量をバイナリパターン化することにより、特徴量数の削減を行い、必要なメモリ量を抑制することが可能となる。しかし、バイナリパターン化することにより、識別時に用いる確率密度分布が疎になる問題が発生する。そこで、学習時に Real AdaBoost を用いてバイナリパターンの統合を行い、密な確率密度分布を作成する。提案手法の有効性を確認するために、人の識別実験と処理に必要なメモリ量の比較を行う。その結果、HOG 特徴量と同程度の識別精度を維持し、処理に必要なメモリ量を削減することができた。

A Method for Reducing number of HOG Features based on Real AdaBoost

CHIKA MATSUSHIMA,^{†1} YUJI YAMAUCHI,^{†1}
TAKAYOSHI YAMASHITA^{†1,†2} and HIRONOBU FUJIYOSHI^{†1}

We propose an efficient method for reducing the number of HOG features based on Real AdaBoost. The proposed method can reduce the amount of memory required by converting the HOG features used in human detection to a binary pattern. Converting to a binary pattern, however, causes the problem of a sparse probability density distribution used in classification. Nevertheless, a dense probability density distribution can be achieved by using Real AdaBoost to integrate the binary pattern during training. To confirm the effectiveness of the proposed method, we conducted evaluation experiments and compared the amounts of memory required. The results show that the detection accuracy of the HOG features is retained, but less memory is required for the processing.

1. はじめに

近年、監視映像からの不審者の発見や ITS における運転手の支援のために、カメラ画像から自動的に人を検出する技術が期待されている。従来、Edge of Orientation Histograms (EOH) 特徴量¹⁾ や Histograms of Oriented Gradients (HOG) 特徴量²⁾ などの局所特徴量を用いた人検出が提案されている²⁾⁻⁷⁾。これらの特徴量は特徴量数が多く、必要なメモリ量が膨大となるという問題があるため、Large Scale Integration (LSI) や Field Programmable Gate Array (FPGA) のような小規模なハードウェアで人検出を実現するには、特徴量数を削減することによりメモリ量を少なくする必要がある。

特徴量数を削減する従来法としては、特徴量のベクトル量子化⁸⁾ や特徴量の次元圧縮を行う主成分分析などがある。ベクトル量子化は、LBG アルゴリズムなどにより入力データを複数のクラスタに分類し、プロトタイプを特徴量とする手法である。主成分分析は、主成分で構成される部分空間に射影することにより特徴量の次元数を削減する手法である。人検出では、画像中の異なる人の大きさに対応するために、検出ウィンドウのサイズを変えながら網羅的にラスタスキャンする必要がある。そのため、膨大な数の検出ウィンドウから特徴量を抽出し、ベクトル量子化や部分空間に射影する主成分分析を行うことは、多くの計算コストが必要となり、小規模なハードウェア構成に向いていない。

そこで、本稿では人検出のための Real AdaBoost^{4),9)} に基づく HOG 特徴量の効率的な削減法を提案する。特徴量の削減には、一般的には実数で表現される HOG 特徴量の勾配強度を、我々は閾値処理によりバイナリパターン化する。しかし、HOG 特徴量をバイナリパターン化することにより、識別時に用いる確率密度分布が疎になるという問題が発生する。そこで、Real AdaBoost を用いて識別器を構築する際に不必要なバイナリパターンの統合を行い、密な確率密度分布の作成を行う。

本稿では、2 章に Real AdaBoost に基づく HOG 特徴量の効率的な削減法について述べ、3 章では提案手法による評価実験の結果を示す。4 章には、高精度化を目的とした異なるセルにおける HOG 特徴量の大小関係を用いたバイナリパターン化と評価実験の結果を示す。

^{†1} 中部大学大学院 工学研究科
Chubu University

^{†2} オムロン株式会社
OMRON Corporation

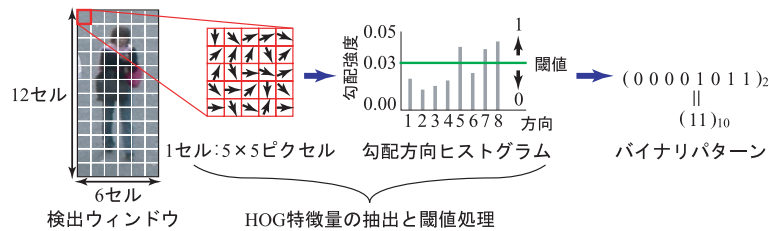


図1 固定閾値によるバイナリパターン化
Fig. 1 Binarization of HOG features by fixed value.

2. Real AdaBoost に基づく HOG 特徴量の効率的な削減法

本章では Real AdaBoost に基づく HOG 特徴量の効率的な削減法について述べる。HOG 特徴量を抽出後、特徴量数とメモリ量を削減するために、固定閾値を用いて HOG 特徴量のバイナリパターン化を行う。バイナリパターン化を行うことにより確率密度分布が疎になるため、提案手法では Real AdaBoost を用いたバイナリパターンの統合を行い、密な確率密度分布を作成する。

2.1 特徴量抽出

提案手法では、図 1 に示すように HOG 特徴量を抽出後、特徴量数とメモリ量を削減するために、固定閾値を用いて HOG 特徴量のバイナリパターン化を行う。

2.1.1 HOG 特徴量

提案手法では、人検出に有効な Histograms of oriented gradients(HOG) 特徴量²⁾を用いる。HOG 特徴量とは、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量であり、物体の形状を表すことが可能である。近接画素の勾配を局所領域によりヒストグラム化するため、照明の影響を受けにくく、局所的な幾何学変化に頑健という特徴がある。はじめに、入力画像をあらかじめ定義したパッチサイズにダウンサンプリングし、各ピクセルの輝度 L から勾配強度 m と勾配方向 θ を次式より算出する。

$$m(x, y) = \sqrt{f_x(x, y)^2 + f_y(x, y)^2} \quad (1)$$

$$\theta(x, y) = \tan^{-1} \frac{f_y(x, y)}{f_x(x, y)} \quad (2)$$

$$\begin{cases} f_x(x, y) = L(x+1, y) - L(x-1, y) \\ f_y(x, y) = L(x, y+1) - L(x, y-1) \end{cases}$$

次に、算出した勾配強度 m と勾配方向 θ を用いてセルの領域ごとに輝度の勾配方向ヒストグラムを作成する。算出される勾配方向は $0^\circ \sim 360^\circ$ となるが、勾配方向の向きを考慮する必要がないため $0^\circ \sim 180^\circ$ に変換する。本研究では変換した勾配方向を 8 方向に分割し、方向ごとに輝度の勾配強度を加算することにより、勾配方向ヒストグラムを作成する。最後に、式 (3) を用いて各セルごとに作成した勾配方向ヒストグラムをブロックごとに正規化し、特徴量を抽出する。

$$v = \frac{v}{\sqrt{\sum_{i=0}^k v_i^2 + \epsilon^2}} \quad (\epsilon = 1) \quad (3)$$

v は HOG 特徴量、 k はブロック内の HOG 特徴量の数を表す。

2.1.2 HOG 特徴量の問題点

HOG 特徴量は、ブロックを移動させながら特徴量を抽出する。そのため、入力画像 30×60 ピクセル、セルサイズ 5×5 ピクセル、ブロックサイズ 3×3 セルの場合、特徴量数は 2,880 個となり、1 個の特徴量は、8 バイト (浮動小数点型) で表現するため、計 23,040 バイトのメモリを必要とする。このとき、画像 1 枚 (720×480 ピクセル) に対し 50,000 個の検出ウィンドウを仮定した場合、必要となるメモリ量は 1.07 GB となる。FPGA や LSI などの小規模なハードウェアでは使用できるメモリ量が少ないため、特徴量数を削減する必要がある。

2.1.3 閾値処理による HOG 特徴量のバイナリパターン化

2.1.1 で抽出した HOG 特徴量数を削減するために、図 1 に示すように、1 セルより得られる 8 方向の HOG 特徴量の勾配強度を閾値処理によりバイナリパターン化する。

$$b_d = \begin{cases} 1 & v_d \geq \text{th} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

b は各勾配方向のバイナリデータ、 $d(1 \sim 8)$ は勾配方向の番号、 th は閾値を表す。HOG 特徴量の抽出と同様に、入力画像 30×60 ピクセル、セルサイズ 5×5 ピクセル、ブロックサイズ 3×3 セルの場合、特徴量数は 360 個となり、1 個の特徴量は 1 バイト (符号なし整数型) で表現可能であるため、浮動小数点で表現した HOG 特徴量と比べ、必要なメモリを 64

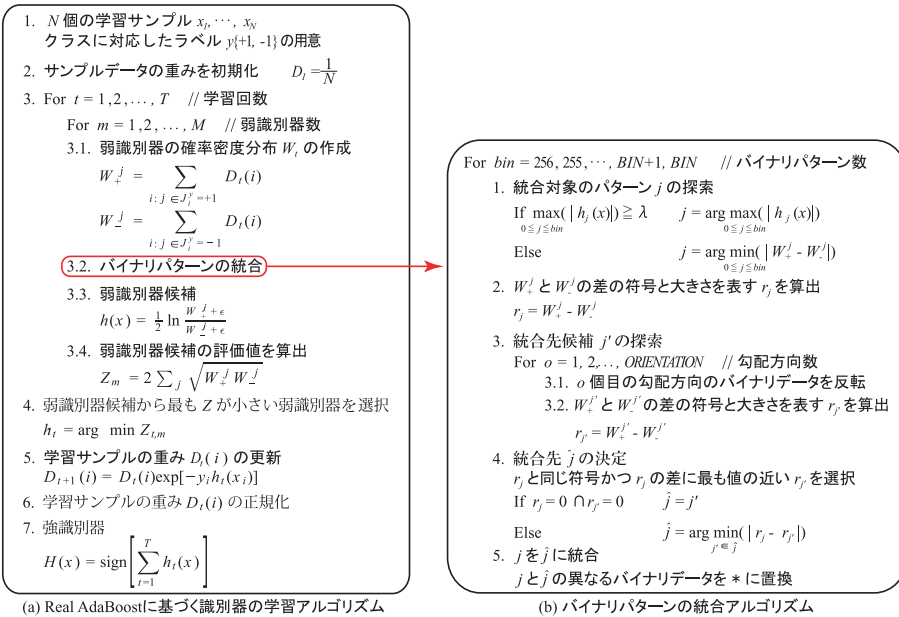


図 2 Real AdaBoost による特徴量の削減アルゴリズム

Fig. 2 Algorithm of reducing number of feature based on Real AdaBoost.

分の 1 に削減することができる。閾値 th は、全セルにおいて同一の閾値を用いるために固定閾値とし、実験より 0.03 とした。

2.2 Real AdaBoost に基づく識別器の学習

バイナリパターン化した特徴量を抽出後、Real AdaBoost により識別器を学習する。アルゴリズムを図 2(a) に示す。まず、図 2(a) の 1. において、ラベルの付いた N 個の学習サンプルを用意し、図 2(a) の 2. において、学習サンプルの重みを初期化する。次に、図 2(a) の 3.1. において、バイナリパターン化した特徴量から人クラス (ポジティブクラス) と非人クラス (ネガティブクラス) の確率密度分布 W_+, W_- を作成する。確率密度分布 W_+, W_- は、1 次元のヒストグラムにより表現され、学習サンプルの重み D_t に基づいて作成する。図 3(a) にバイナリパターン化した特徴量を用いて作成した確率密度分布 W_+, W_- を示す。各特徴量ごとに作成した確率密度分布 W_+, W_- を用いて、弱識別器の出力 $h(x)$ を算出し、弱識別器の候補とする。ここで、 ϵ は分母が 0 になることを防ぐ係数であり、文献⁽⁹⁾ より

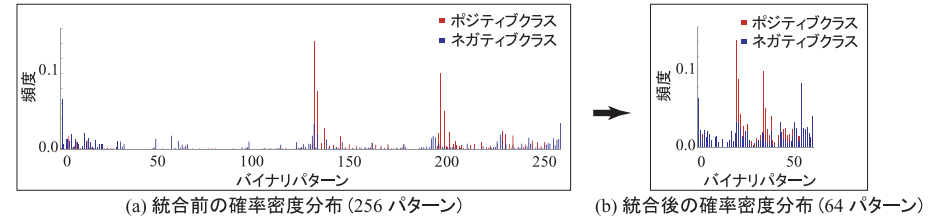


図 3 1 ラウンド目の学習で選択された確率密度分布
Fig. 3 An example of probability density distribution at 1st round training.

$\epsilon = 1/N$ とした。弱識別器の候補を算出した後、図 2(a) の 3.4. において確率密度分布 W_+, W_- の分離度を表す評価値 Z_m を算出する。評価値 Z_m は小さいほど、ポジティブクラスとネガティブクラスの分布が分離していることを意味する。そのため、図 2(a) の 4. において、多数の弱識別器候補から評価値 Z が最小となる確率密度分布を持つ弱識別器を、 t ラウンド目において最も識別に有効な弱識別器として選択する。弱識別器を選択後、図 2(a) の 5. において誤識別した学習サンプルが、次のラウンドで正しく識別されるために学習サンプルの重みを更新する。確率密度分布の作成から学習サンプルの重み更新までの処理を一定の学習回数、または一定の識別率が得られるまで繰り返すことにより、強識別器 $H(x)$ を得る。

2.2.1 HOG 特徴量のバイナリパターン化による問題

バイナリパターン化した特徴量を用いて Real AdaBoost により学習した際に、1 回目の弱識別器として選択された確率密度分布 W_+, W_- を図 3(a) に示す。確率密度分布 W_+, W_- の横軸はバイナリパターンの番号、縦軸は頻度を表す。作成された確率密度分布は、頻度の低いバイナリパターンが多く存在し、その分布は疎であることがわかる。そのため、HOG 特徴量をバイナリパターン化することにより以下の問題が生じる。

問題点 弱識別器が極端に大きな値や小さな値を出力

確率密度分布が疎になると、一方のクラスの頻度が 0 に近いバイナリパターンが存在する。このようなバイナリパターンは、次式より計算した弱識別器の出力 $h(x)$ が極端に大きな値もしくは小さな値となる。

$$h(x) = \frac{1}{2} \ln \frac{W_+ + \epsilon}{W_- + \epsilon} \quad \left(\epsilon = \frac{1}{N} \right) \quad (5)$$

強識別器 $H(x)$ は弱識別器 $h(x)$ の線形和に依存するため、極端に大きな値や小さな値

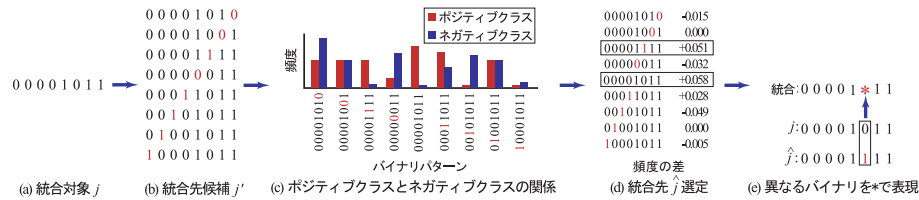


図 4 統合対象の探索例
Fig. 4 Example of searching for integration pattern.

は誤識別の要因となる可能性がある．この問題を解決するには，多量の学習サンプルを準備して疎な分布を埋めることが考えられるが，実際には密な分布となるように多量の学習サンプルを準備することは困難である．

2.2.2 Real AdaBoost に基づくバイナリパターンの統合

2.2.1 で述べた問題を解決するために，図 2(b) のアルゴリズムに示したバイナリパターンの統合を行い，図 3(b) に示すような密な確率密度分布を作成する．確率密度分布 W_+ , W_- が疎になると，弱識別器が極端に大きな値や小さな値を出力するという問題がある．そのため，一方のクラスの頻度が 0 に近い値であるバイナリパターンを統合する．さらに，一方のクラスの頻度が 0 に近い値のバイナリパターンは限りがあるため，ポジティブクラスとネガティブクラスの確率の差が小さく，識別に不必要なバイナリパターンを統合する．これにより，密な分布を作成し，誤識別を抑制する．統合手順を以下に示す．

バイナリパターンの統合を行うために，図 2(b) の 1. に示すように統合対象のバイナリパターン j を選択する．この際に用いる閾値 λ は実験より 2.0 とする．統合対象のバイナリパターン j を決定した後，図 2(b) の 3. に示す統合先のバイナリパターン候補 j' を探索する．そこで図 4(b) に示すように，右から順に 1 ビットだけバイナリデータを反転し，8 通りの統合先候補 j' を探索する．統合対象 j と統合先候補 j' のポジティブクラスとネガティブクラスの差を表す r を算出し，図 2(b) の 4. において，クラス間の差 r が類似している統合先 \hat{j} を決定し，統合対象 j と統合先 \hat{j} の各クラスの頻度を加算する．ここで，統合対象と統合先のバイナリパターンにおいて異なるバイナリデータを“*”で表現する．“*”で表現したバイナリデータは識別に不要であるため，識別時の演算を削減することができる．以上の統合処理を一定のパターン数になるまで繰り返す．

提案する Real AdaBoost によるバイナリパターンの統合により，図 3(b) に示すような密な分布を作成することができる．これにより，図 5 に示すように，誤識別の要因となる弱

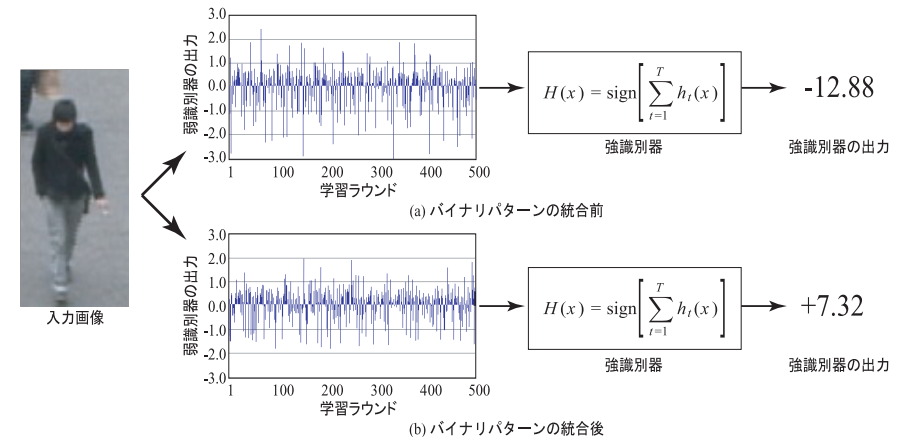


図 5 強識別器の出力
Fig. 5 Examples of output of strong classifier.



図 6 データベースの例
Fig. 6 Example of database.

識別器の出力を抑制することができる強識別器 $H(x)$ の結果を得ることが可能となる．

3. 評価実験

提案手法の有効性を確認するために評価実験を行う．

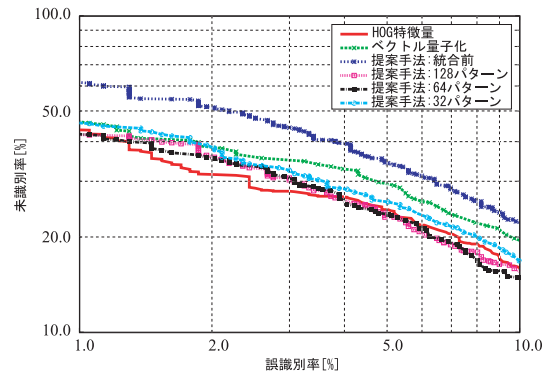


図 7 評価実験による DET カーブ
Fig. 7 DET curves of evaluation experiments.

3.1 データベース

評価に用いる人のデータベースの一部を図 6 に示す。学習用サンプルとして、ポジティブクラス 2,054 枚、ネガティブクラス 6,258 枚、評価用サンプルとして、ポジティブクラス 1,024 枚、ネガティブクラス 1,234 枚を用いる。

3.2 実験概要

提案手法の有効性を確認するために、従来法である HOG 特徴量、特徴量を削減する手法として用いられるベクトル量子化と比較する。各手法において 400 回の学習を行い、識別精度の評価と使用するメモリ量の比較を行う。評価には、誤識率と未識別率からなる Detection Error Tradeoff (DET) カーブを用いる。DET カーブは原点に近いほど性能が良いことを示す。また、画像 1 枚 (720 × 480 ピクセル) に対し 50,000 個の検出ウィンドウを仮定し、識別に必要な特徴のメモリ量を比較する。メモリ量は、特徴量のデータ型 × 特徴量数 × 検出ウィンドウ数より算出する。

3.3 実験結果

図 7 に実験結果の DET カーブ、表 1 に誤識率 5.0% における識別率と識別に必要なメモリ量を示す。このとき、バイナリパターンの統合数は、 2^n ($n = 5, 6, 7$) とする。図 7 より、提案手法は HOG 特徴量のバイナリパターン化やベクトル量子化よりも原点に近いことから精度がよいことがわかる。表 1 より、64 パターンになるまでバイナリパターンを統合した提案手法は、HOG 特徴量と同程度の識別率を維持したままメモリ量を約 98.0%削減す

表 1 各手法の識別率と画像 1 枚を処理するために必要なメモリ量

Table 1 Classification rate and memory size for processing an image.

手法		識別率 [%]	メモリ量 [MB]	
従来法	HOG 特徴量 ²⁾	75.6	1,098.63	
	ベクトル量子化	70.8	198.75	
提案手法 (バイナリパターン)	統合前	65.8	17.17	
	統合後	128 パターン		76.5
		64 パターン		76.5
		32 パターン		74.2

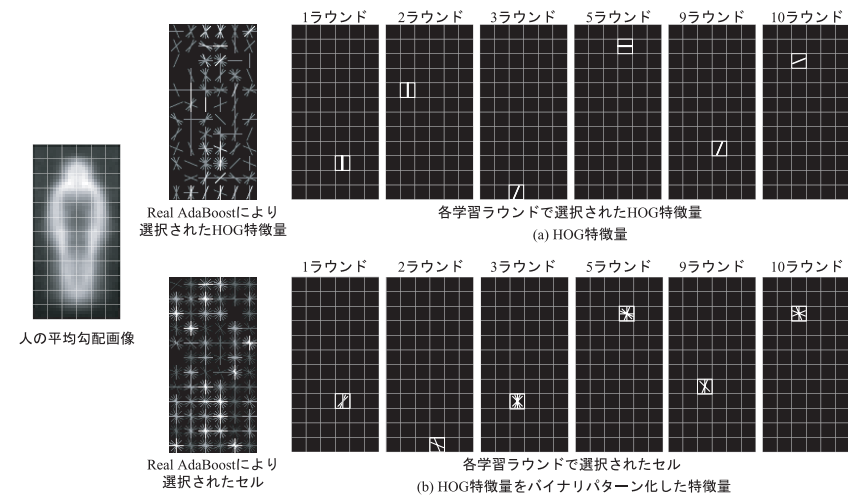


図 8 学習において選択された特徴量の可視化
Fig. 8 Visualization examples of selected feature by training.

ることができた。

3.4 考察

評価実験において Real AdaBoost により選択された HOG 特徴量と HOG 特徴量をバイナリパターン化した特徴量の可視化例を図 8 に示す。このとき、輝度が高いほど識別に有効な特徴量であることを表わす。図 8(a) に示した HOG 特徴量の可視化結果より、人の形状を捉えるセルが多く選択されている。図 8(b) に示したバイナリパターン化した特徴量の可視化結果より、HOG 特徴量と同様に、人の形状を捉えるセルが多く選択されている。ま

た、各特徴量の初期ラウンドで選択された可視化結果より、類似した勾配方向の特徴量が選択されていることがわかる。以上より、特徴量をバイナリ化しても、HOG 特徴量と同様の特徴表現ができているといえる。

4. 異なるセルの HOG 特徴量の大小関係を用いたバイナリパターンによる高精度化

Real AdaBoost に基づくバイナリパターンの統合を行うことにより、従来の識別率を維持したまま必要なメモリ量を削減することができた。人検出の高精度化には、異なる2つのセルを同時に捉えることの有効性が報告されている¹⁰⁾。そこで本章では、異なるセルにおける HOG 特徴量の大小関係を用いたバイナリパターンの作成を行い、人検出能力の高精度化について検討する。

4.1 バイナリパターンの作成

2.1.3 に示した HOG 特徴量のバイナリパターン化では、最適な閾値を決定する必要がある。しかし、学習サンプルと実際に人検出を行う環境が異なる場合では、最適な閾値を決定することは困難である。そこで、図9に示すように異なるセルから得られる8方向の HOG 特徴量の大小関係を用いて、式(6)よりバイナリパターンを作成する。

$$b_d = \begin{cases} 1 & v_d \geq v_{d'} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

ここで、 b は勾配方向のバイナリデータ、 d はセルの勾配方向、 v はセルにおける各勾配方向の HOG 特徴量を表す。図9に示すように、異なるセルにおける HOG 特徴量の大小関係を用いることにより HOG 特徴量の対称性を表現することが可能となる。しかし、図10に示すように2つのセルの HOG 特徴量が類似している場合、図9に示す方法では、HOG 特徴量の大小関係を明確に表現することができない。そのため、バイナリパターンを作成する際に、一方のセルから抽出される HOG 特徴量の勾配方向をビットシフトすることにより、2つのセルにおける HOG 特徴量の大小関係が明確なバイナリパターンを作成する。学習時に確率密度分布が疎になる問題を解決するために、バイナリパターンを作成した後、2.2.2 に示した Real AdaBoost に基づくバイナリパターンの統合を行う。

4.2 評価実験

異なるセルにおける HOG 特徴量の大小関係を用いたバイナリパターン化の有効性を確認する評価実験を行う。

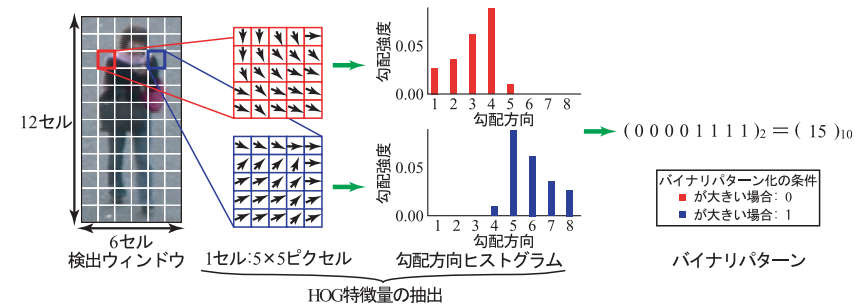


図9 異なるセルの HOG 特徴量を用いたバイナリパターン
Fig. 9 Binary pattern using HOG feature of different cells.

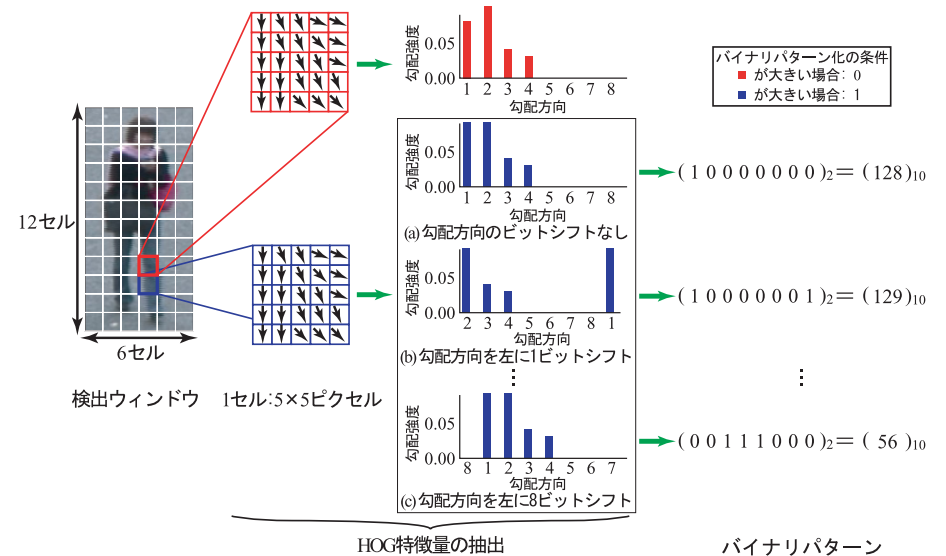


図10 勾配方向のビットシフトを含むバイナリパターン
Fig. 10 Binary pattern including bit shift of orientation.

4.2.1 実験概要

異なるセルにおける HOG 特徴量の大小関係を用いたバイナリパターン化の有効性を確認するために、固定閾値によるバイナリパターン化、異なるセルにおける HOG 特徴量の大小関係を用いたバイナリパターン化と比較する。提案手法における統合後のバイナリパター

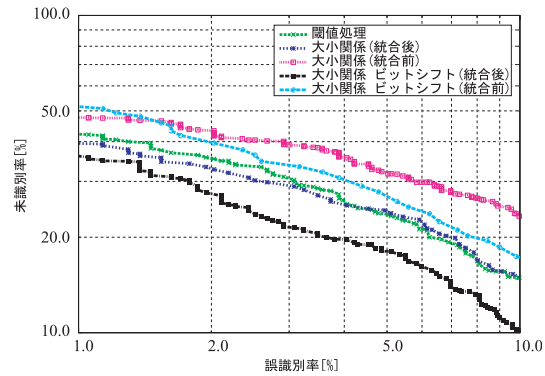


図 11 バイナリパターン化した特徴量の評価実験による DET カーブ
Fig. 11 DET curves of evaluation experiments by binarization of HOG features.

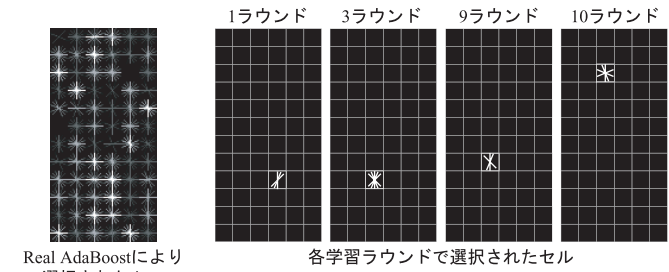
ン数は、3章の評価実験より最適な統合数である 64 パターンとした。各手法において 400 回の学習を行い、DET カーブにより評価する。

4.2.2 実験結果

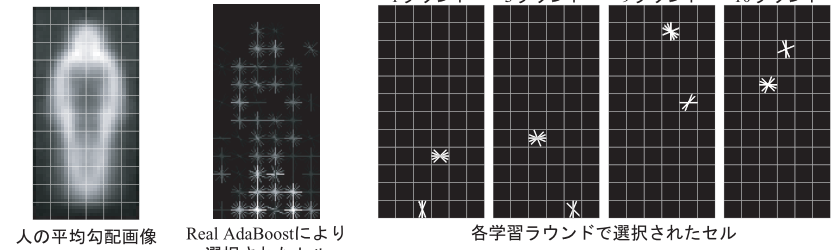
図 11 に実験結果の DET カーブ、表 2 に誤識別率 5.0% における識別率を示す。図 11 より、異なるセルにおける HOG 特徴量の大小関係によるバイナリパターン化は、閾値処理によるバイナリパターン化と同様に、統合することで識別精度が向上することがわかる。また、HOG 特徴量の大小関係を用いてバイナリパターン化する際に、HOG 特徴量の勾配方向をビットシフトすることにより高精度になる。表 2 より、誤識別率 5.0% における識別率を比較した場合、閾値によるバイナリパターン化は 76.5% であるのに対し、異なるセルにおける HOG 特徴量の大小関係によるバイナリパターン化 (ビットシフト有) は 82.0% であるため、識別率を 6.4% 向上させることができた。

4.2.3 考察

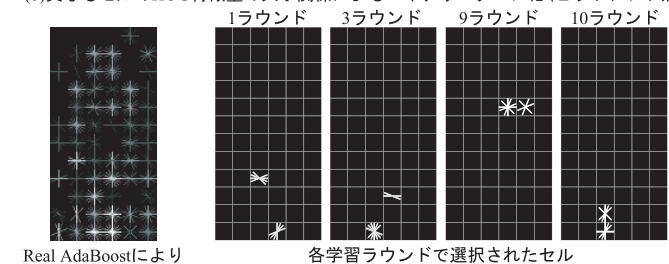
図 12 に学習において選択された特徴量の可視化例を示す。図 12(b) は、2 つのセルが人の形状に沿った HOG 特徴量が選択されている。また、図 12(c) は、図 12(b) で選択された特徴量を含み、初期のラウンドでは隣り合うセルの特徴量が選択されている。図 12(b) と (c) において学習回数が 400 回である場合、隣り合うセルの特徴量が選択される割合を算出すると、図 12(b) は 4.0% であるのに対し、図 12(c) は 12.0% である。以上より、HOG 特徴量の勾配方向の大小関係を用いることにより人の形状を捉えることができ、ビットシフト



(a) 固定閾値による HOG 特徴量のバイナリパターン化



(b) 異なるセルの HOG 特徴量の大小関係によるバイナリパターン化 (ビットシフト無)



(c) 異なるセルの HOG 特徴量の大小関係によるバイナリパターン化 (ビットシフト有)

図 12 学習において選択された特徴量におけるバイナリパターンの可視化
Fig. 12 Visualization example of selected binary pattern by training.

表 2 各手法の識別率
 Table 2 Classification rate for each method.

手法		識別率 [%]
固定閾値による閾値処理		76.5
異なるセルの大小関係	統合前	68.3
	統合後	76.5
異なるセルの大小関係 (ビットシフト有)	統合前	74.9
	統合後	82.0

を行うことで隣り合うセルを選択することができた。そのため、ビットシフトを行うことで特徴量の表現能力が向上したといえる。

5. おわりに

本稿では、人検出のための Real AdaBoost に基づく HOG 特徴量の効率的な削減法について提案した。人検出で用いられる HOG 特徴量は特徴量数が多いため、HOG 特徴量をバイナリパターン化することにより、HOG 特徴量の削減を行った。しかし、バイナリパターン化することにより識別時に用いる確率密度分布が疎になるため、Real AdaBoost を用いて学習をする際に不必要なバイナリパターンの統合を行い、密な確率密度分布を作成した。提案手法の有効性を確認するために、人画像のデータベースを用いて評価を行い、従来法の精度を維持したままメモリ量を約 98.0%削減することができた。また、HOG 特徴量の大小関係を用いることにより、閾値処理を行うことなくバイナリパターンを作成することができ、識別率を 6.4%向上させることができた。今後は、リアルタイム人検出について検討する予定である。

参 考 文 献

- 1) Levi, K. and Weiss, Y.: Learning Object Detection from a Small Number of Examples: the Importance of Good Features, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.53–60 (2004).
- 2) Dalal, N. and Triggs, B.: Histograms of Oriented Gradients for Human Detection, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol.1, pp.886–893 (2005).
- 3) Chen, Y.T. and Chen, C.S.: Asian Cascade of Feed-Forward Classifiers for Fast Pedestrian Detection, *Eighth Asian Conference on Computer Vision*, pp.905–914 (2007).
- 4) Wu, B. and Nevatia, R.: Detection of Multiple, Partially Occluded Humans in a

- Single Image by Bayesian Combination of Edgelet Part Detectors, *IEEE International Conference on Computer Vision*, pp.90–97 (2005).
- 5) Mu, Y., Yan, S., Liu, Y., Huang, T. and Zhou, B.: Discriminative local binary patterns for human detection in personal album, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.1–8 (2008).
- 6) Munder, S. and Gavrila, D.M.: An Experimental Study on Pedestrian Classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.1863–1868 (2006).
- 7) Hou, C., Ai, H. and Lao, S.: Multiview Pedestrian Detection Based on Vector Boosting, *Eighth Asian Conference on Computer Vision*, pp.220–229 (2007).
- 8) Linde, Y., Buzo, A. and Gray, R.: An Algorithm for Vector Quantizer Design, *IEEE Transactions on Communications*, Vol.28, No.1, pp.84–95 (1980).
- 9) Schapire, R.E. and Singer, Y.: Improved Boosting Algorithms Using Confidence-rated Predictions, *Machine Learning*, pp.297–336 (1999).
- 10) 三井相和, 山内悠嗣, 藤吉弘巨: Joint HOG 特徴を用いた 2 段階 AdaBoost による人検出, 第 14 回画像センシングシンポジウム SSII08 (2008).