

# EtC 画像からの HOG 特徴量抽出とその機械学習による画像分類への応用

北山 昌希<sup>†</sup> 貴家 仁志<sup>†</sup>

<sup>†</sup> 首都大学東京

〒 191-0065 東京都日野市旭丘 6-6

あらまし 本稿では、画像の視覚的情報が保護された EtC(Encryption-then-Compression) 画像から HOG 特徴量を抽出して、その特徴量を各種機械学習法に適用する手法を提案する。ここで、EtC 画像とは、JPEG 圧縮を可能とするブロックベース暗号化が施された画像である。近年、クラウドサービスを利用し、プロバイダーの提供する計算資源を利用する計算形態が急速に普及している。しかし、プロバイダーの信頼性欠如や事故によって、データの不正利用、流出、プライバシーの侵害などの問題が危惧されている。本稿ではこのような背景から、プライバシーの保護された画像からの HOG 特徴量を抽出する手法を考察する。まず、微分画像生成を、暗号化のブロック単位で行い、同一鍵を画像間で用いた場合、暗号化処理が HOG 特徴量に影響を与えないことが示される。さらに、抽出された HOG 特徴量を用いて、線形サポートベクターマシン (SVM)、ガウシアンカーネル SVM、決定木、1 分割決定木を弱分類器とする AdaBoost の 4 つの分類器を学習し、提案法の有効性を確認する。

キーワード Encryption-then-Compression, HOG, SVM, 決定木, AdaBoost

## HOG feature extraction from EtC images, and its application to image classification with machine learning

Masaki KITAYAMA<sup>†</sup> and Hitoshi KIYA<sup>†</sup>

<sup>†</sup> Tokyo Metropolitan University

Asahigaoka 6-6, Hino-shi, Tokyo, 191-0065

**Abstract** In this paper, we propose a scheme of HOG feature extraction from EtC(Encryption-then-Compression) images without visual informations, and apply the features to various machine learning algorithms, where EtC images, which can be compressed by JPEG compression, are images encrypted by a block-based encryption method. Recently, cloud computing is spreading in many fields. However, the cloud computing has some serious issues for end users, such as unauthorized use and leak of data, and privacy compromise, due to unreliability of providers and some accident. Because of such a situation, we consider a scheme of HOG feature extraction from privacy-preserving images. First, we show that the encryption process does not effect on the HOG features if differential operations are carried out within each block of the encryption and the same secret key is commonly used for all images. Then, by using the extracted HOG features, four classifiers: linear Support Vector Machine(SVM), gaussian SVM, decision tree, and AdaBoost with one splitted decision trees as weak classifiers, are trained to confirm the effectiveness of the proposed method.

**Key words** Encryption-then-Compression, HOG, SVM, decision tree, AdaBoost

### 1. ま え が き

近年、様々な計算分野において、プロバイダーの計算資源を利用するクラウドコンピューティングやエッジコンピューティングが急速に普及してきている。しかしクラウドコンピューティングの利用は、プロバイダーの信頼性を前提にしており、その信頼性の欠如や事故によって、データの不正利用や流出、プライバ

シーの侵害といった問題の発生が危惧されている [1]。今後のクラウドコンピューティングの普及にとって、データの不正利用や流出、エンドユーザーのプライバシー問題をいかに解決するかが重要な課題となっている。このような背景から、本稿ではクラウドサービス上での機械学習による画像分類サービスを想定し、プライバシーを保護した EtC(Encryption-then-Compression) 画像からの HOG 特徴量抽出法を考察する。

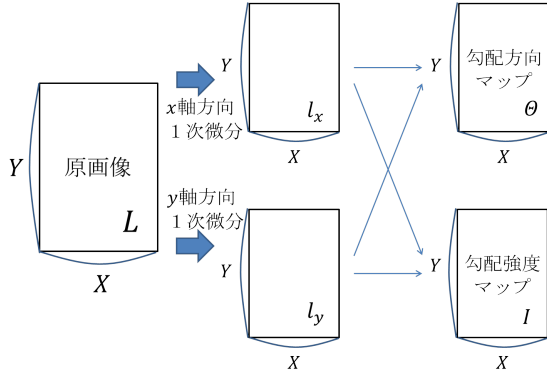


図1 勾配方向マップおよび勾配強度マップの取得手順.

本稿における EtC 画像とは、JPEG 圧縮の使用を前提として提案されたブロックベース暗号化処理が施された画像である [2]. EtC 画像は、データ圧縮された形式で保存可能であり、かつ総当たり攻撃やジグソーパズル開放攻撃などに対して安全性がすでに評価されている [3]. HOG 特徴量は、画像の局所領域ごとの輝度勾配ヒストグラムを特徴ベクトルとするものである. HOG 特徴量は画像の拡大縮小にロバストであるという強みを有し、パターン認識の分野で広く用いられている特徴量である.

本稿では、画像データベースおよびクエリデータに対して、同一鍵を用いて EtC 変換を施し、HOG 特徴量を抽出することを考える. この時、EtC 変換による同一のブロック内においては、EtC 変換は HOG 特徴量抽出に影響を与えないことを示す. 次に、EtC 画像全体から HOG 特徴量を抽出するために、ブロックごとに微分画像を生成し、それらを連結したものを EtC 画像全体の微分画像とする手法を提案する.

最後に、提案法の妥当性を、原画像から抽出された HOG 特徴量との比較実験によって評価する. 本実験では、線形 SVM、ガウシアンカーネル SVM、決定木、1 分割決定木を弱分類器とした AdaBoost の 4 つの分類器を学習し、提案法の有効性を確認する.

## 2. 準備

### 2.1 HOG 特徴量

Histograms of Oriented Gradients(HOG) [4] は、画像の局所領域ごとに画素値の輝度勾配ヒストグラムから抽出される特徴量である. 画像の拡大縮小にロバストな特徴量であり、パターン認識の分野を中心として広く用いられている. 以下に抽出手順を簡単に示す.

#### 2.1.1 勾配強度及び勾配方向

HOG 特徴量の抽出では、まず原画像から勾配方向マップおよび勾配強度マップを取得する. 図 1 はその計算フローである. 入力画像  $L$  (サイズ  $X \times Y$ ) を式 (1) および式 (2) により 1 次微分し、 $x$  軸方向の微分画像  $l_x$  及び  $y$  軸方向の微分画像  $l_y$  をそれぞれ得る.

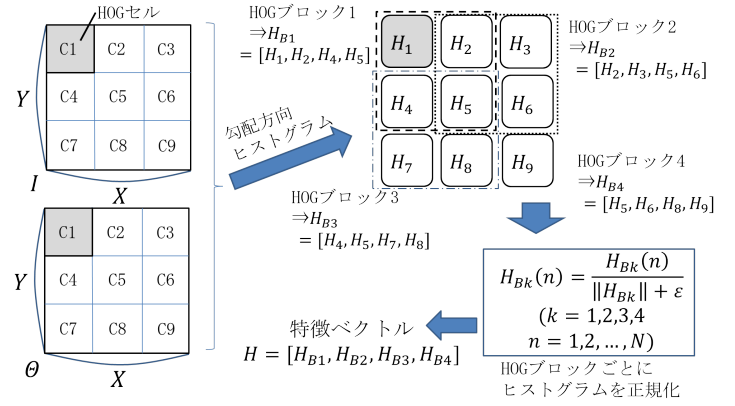


図2 HOG 特徴量の取得

$$l_x(x, y) = \begin{cases} L(x+1, y) - L(x-1, y) & (2 \leq x \leq X-1) \\ L(2, y) - L(1, y) & (x = 1) \\ L(X, y) - L(X-1, y) & (x = X) \end{cases} \quad (1)$$

$$l_y(x, y) = \begin{cases} L(x, y+1) - L(x, y-1) & (2 \leq y \leq Y-1) \\ L(x, 2) - L(x, 1) & (y = 1) \\ L(x, Y) - L(x, Y-1) & (y = Y) \end{cases} \quad (2)$$

ここで、 $L(x, y), x \in \{1, 2, \dots, X\}, y \in \{1, 2, \dots, Y\}$  は座標  $(x, y)$  における  $L$  の画素値である. 次に、得られた微分画像を用いて式 (3) および式 (4) に従い勾配強度マップ  $I$  及び勾配方向マップ  $\theta$  をそれぞれ座標  $(x, y)$  毎に求める.

$$I(x, y) = \sqrt{l_x^2(x, y) + l_y^2(x, y)} \quad (3)$$

$$\theta(x, y) = \tan^{-1} \frac{l_y(x, y)}{l_x(x, y)} \quad (4)$$

#### 2.1.2 HOG セルと勾配方向ヒストグラム

図 2 に例示するように、勾配方向マップ  $\theta$  と勾配強度マップ  $I$  をそれぞれ局所領域に分割し、HOG セルとする. HOG セルごとに勾配強度  $I$  で重みづけた勾配方向ヒストグラムを求め、次に、複数の HOG セルをまとめた HOG ブロックを定義し、HOG ブロックごとに勾配方向ヒストグラムを連結し、正規化処理を施す. 最後に全 HOG ブロック分のヒストグラムを連結して特徴ベクトルとする.

図 2 の例では、画像を 9 つの HOG セルに分割し、HOG セル毎に勾配方向ヒストグラム  $H_1, H_2, \dots, H_9$  を求めている. その後、4 つの HOG セルをまとめ HOG ブロックとし、中間 HOG セルの重複を許容して計 4 つの HOG ブロックを定義する. 各 HOG ブロック毎に 4 つのヒストグラムを連結したヒストグラム  $H_{B1}, H_{B2}, H_{B3}, H_{B4}$  を得る. 最後に、 $H_{B1}, H_{B2}, H_{B3}, H_{B4}$  を式 (5) に従い正規化し、それらを連結してその画像の特徴ベクトル  $H$  とする.

$$H_{Bk}(n) = \frac{H_{Bk}(n)}{\|H_{Bk}\| + \epsilon} \quad k = 1, 2, 3, 4 \quad (5)$$

$$n = 1, 2, \dots, N$$

ただし、 $N$  は HOG ブロックのヒストグラムにおけるビン数で

あり,  $H_{Bk}(n)$  はヒストグラム  $H_{Bk}$  の  $n$  番目のビンの度数である.  $\varepsilon$  は分母が 0 となることを回避するための微小な定数である.

## 2.2 サポートベクターマシン (SVM)

SVM [5] は機械学習における教師あり学習の一種である. 2 値分類アルゴリズムであり, 特徴空間を「マージン最大化」を行いながら線形分割する.

今, 入力ベクトル  $\mathbf{x}$  が正例なら正の値, 負例なら負の値を出力する線形識別関数  $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} - b$  (6) を考える. SVM では, 超平面  $f(\mathbf{x}) = 0$  が各教師ベクトルからできるだけ遠くなるようにパラメータ  $\mathbf{w}$ ,  $b$  を学習する. これが「マージン最大化」の戦略である. SVM におけるマージン最大化のためのパラメータ学習は, 次の双対問題を  $\alpha$  について解くことに帰着する.

$$\begin{aligned} \max. \quad & \left\{ -\frac{1}{2} \sum_{j,j} \alpha_i \alpha_j y^{(i)} y^{(j)} \mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)} + \sum_i \alpha_i \right\} \\ \text{s.t.} \quad & \sum_i \alpha_i y^{(i)} = 0, \\ & 0 \leq \alpha_i \leq C; \forall i. \end{aligned} \quad (7)$$

ただし  $y$  はクラスラベルであり, 正例の場合は 1, 負例の場合は -1 である. また,  $C$  は正則化係数である. この式より, 教師ベクトル  $\mathbf{x}$  は問題中において常に内積の形で現れることがわかる. これをカーネル法と言い, 式 (7) の内積部を一般化してカーネル関数  $K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  と表す. カーネル関数には, 単純な教師ベクトル間の内積以外に様々な種類がある, 例えば, 多項式カーネル  $K_p$ , ガウシアンカーネル  $K_g$  は,

$$\begin{aligned} K_p(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) &= (\mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)} + r)^d \\ K_g(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) &= \exp(-\gamma |\mathbf{x}^{(i)} - \mathbf{x}^{(j)}|^2) \end{aligned}$$

と表現される. ただし  $r, d, \gamma$  はハイパーパラメータである. これらのカーネル関数は, 元の特徴ベクトルを高次元に拡張していることと等価であり, SVM の非線形分類を可能としている. これをカーネルトリックという.

## 2.3 決定木

決定木 [6] は古典的な教師あり学習アルゴリズムであり, 分割条件によって再帰的に教師データを分割していくことで学習を行う. 決定木は各分割をノードとした木構造であり, 根ノードに入力された新規ベクトルは分割条件を適用しながらノードをたどり, 葉ノードによりクラスラベルを出力する. 決定木の学習の詳細を簡単に要約する.

### 2.3.1 ノードにおける分割の進み具合の定義

まず, ノードにおける教師ベクトルの集合から, そのノードにおける分割の進み具合を測るための指標を定義する. この指標には複数の種類があるが, ここでは代表的なジニ指数について述べる. ノード  $t$  において全部で  $n$  個の教師データがあり, その内クラス  $k$  のデータが  $n_k$  個存在するとする. このとき, ノード  $t$  においてクラス  $k$  が生起する確率  $p_t(k)$  とすると, ノード  $t$  のジニ指数  $I_{G,t}$  は,

$$I_{G,t} = 1 - \sum_k p_t^2(k)$$

ただし,

$$p_t(k) = \frac{n_k}{n}$$

と表現される. よく分類されているノードほどジニ指数は小さくなり, ノードに含まれるクラスが 1 つしかない場合 0 となる.

### 2.3.2 各ノードにおける分割条件の決定

分割条件を決定したいノードを  $t_1$ , 分割後のノードを  $t_2, t_3$  とすると, 各ノードのジニ指数  $I_{G,t_1}, I_{G,t_2}, I_{G,t_3}$  から, ノード  $t_1$  における情報利得  $IG(t_1)$  を次式より求める.

$$IG(t_1) = I_{G,t_1} - \frac{N_2}{N_1} I_{G,t_2} - \frac{N_3}{N_1} I_{G,t_3} \quad (8)$$

ただし  $N_1, N_2, N_3$  は各ノードにおけるデータ数である. ノード  $t_1$  における  $i$  番目の特徴ベクトル  $\mathbf{x}^{(i)}$  における  $j$  次元目の要素を  $a_{i,j}$  とすると, 全ての  $a_{i,j}$  についてこれを閾値とした分割条件を適用し, 最も情報利得が大きかったものを採用する. この分割プロセスをある制約下で再帰的に繰り返すことで決定木が学習される.

## 2.4 AdaBoost

AdaBoost [7] は機械学習におけるアンサンブル学習アルゴリズムの一種である. アンサンブル学習は多数の分類器 (弱分類器) を組み合わせることで 1 つの分類器 (強分類器) を構成する手法である. AdaBoost は, 前の弱分類器の誤判別を修正するように逐次的に弱分類器を構成していく手法であり, 決定木等のアルゴリズムと組み合わせることで性能を改善させるために広く用いられている.

AdaBoost のアルゴリズムの概略を以下に示す.

- 1 教師ベクトルに均等に重みを付与.
- 2 教師ベクトルの重みを用いて  $t$  番目の弱分類器  $V_t$  を学習.
- 3  $V_t$  の重み付き分類誤差の期待値  $\varepsilon_t$  を計算. それを元に信頼度  $\alpha_t$  を計算.
- 4 教師ベクトルの重みを更新.  $V_t$  が誤判別した教師ベクトルの重みを大きくし, 正規化する. 手順 2 に戻るか, 手順 5 へ.
- 5 弱分類器  $V_t (t = 1, 2, \dots, T)$  を信頼度  $\alpha$  で重み付き多数決をとり, 最終的な強分類器とする.

## 3. 提案法

### 3.1 画像のブロックベース暗号化 (EtC)

静止画像の暗号化法として, 画像をブロックに分割して処理を行うブロックベース暗号化が研究されている [8]. この暗号化法は, 画像の JPEG 圧縮前に適用可能であるという特徴を有し, この暗号化が適用された画像を EtC (Encryption then Compression) 画像と呼ぶ. 本稿では, 画像を EtC 画像に変換し, EtC 画像に対して機械学習法を適用することを考察する. 以下に白黒画像から EtC 画像を生成するための 3 つのステップを要約する.

#### A. ブロックスクランブル

まず, サイズ  $X \times Y$  の画像を一定サイズ  $B \times B$  のブロックに分割する. ブロックスクランブルは, このブロックを乱数を用いてランダムに置換する操作である. 図 3 に, 原画像をブロック分割し, ブロックスクランブルを施す例を示す.

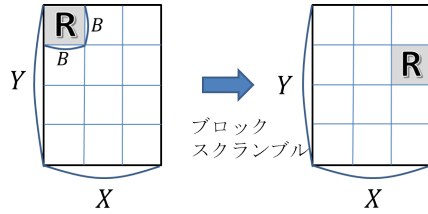


図3 画素のブロック分割とブロックスクランブルの例.

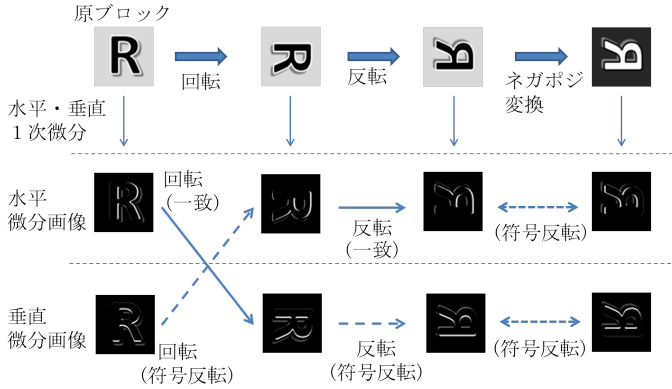


図4 ブロック内変換と、対応する微分画像の例.

### B. ブロックの回転・反転

ブロックスクランブルの後に、各ブロックに対しランダムに回転および反転変換を施す。ブロックの回転変換は、各ブロックを $0^\circ$ 、 $90^\circ$ 、 $180^\circ$ 、 $270^\circ$ の4つのいずれかの角度だけランダムに回転させる操作である。ブロックの反転変換は、ブロック内の画素を水平または垂直方向にランダムで反転させる方法である。反転しない、水平方向のみ反転、垂直方向のみ反転、水平・垂直両方向の反転の4つのパターンがある。ただし、水平・垂直両方向の反転変換は $180^\circ$ の回転変換と等しい。よって、回転・反転変換を合わせた変換の総パターンは12通りである。

### C. ネガポジ変換

ネガポジ変換は、ランダムにブロックを選択して、選択されたブロック内の全ての画素値を反転させる方法である。ブロック $i$ 内の画素値を $p$  ( $0 \leq p \leq 255$ )、鍵 $K$ により生成された2値の乱数を $r(i)$ としたとき、次式によりネガポジ変換が実行される。

$$\begin{cases} p' = p & (r(i) = 0) \\ p' = 255 - p & (r(i) = 1) \end{cases} \quad (9)$$

ただし $r(i)$ の発生確率は $p(r(i)) = 0.5$ である。ネガポジ変換を行うことにより、画像の濃淡やヒストグラムが変化し、より原画像の特定が困難になる。

## 3.2 EtC 画像の HOG 特徴量

本稿では、白黒画像の $J$ 個の教師データベース $L_j, j = 1, 2, \dots, J$ に対して同一鍵を用いて EtC 変換を施し、EtC 画像から HOG 特徴を抽出する。

図4では、EtC 画像生成のための各ステップが、微分画像の計算にどのような影響を与えるかを、1つのブロックに着目し

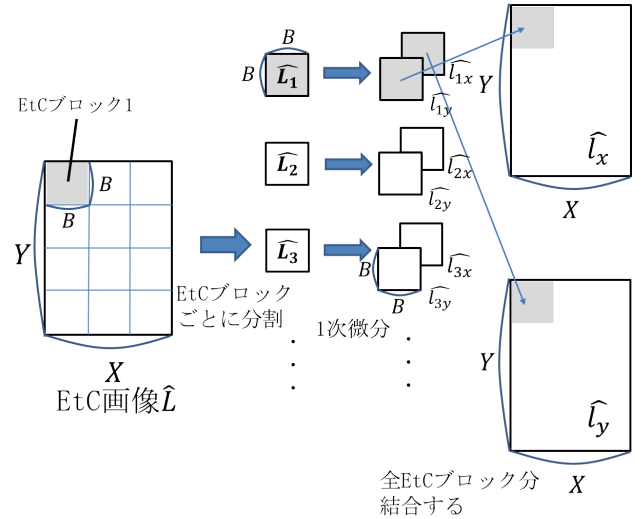


図5 EtC 画像全体からの微分画像の生成手法

として例示している。例えば、原ブロックの微分画像(水平)を $90^\circ$ 右へ回転させた画像は、原ブロックを $90^\circ$ 右に回転させた画像の微分画像(垂直)に一致する。一方、原ブロックの微分画像(垂直)を $90^\circ$ 右へ回転させた画像は、原ブロックを $90^\circ$ 右に回転させた画像の微分画像(水平)とは画素値を符号反転した関係にある。図4は、このような関係を例示したものである。EtC 画像生成のための各ステップと、原画像の微分画像との関係・組合せは、図4以外にも存在する。すべての組み合わせにおいて、EtC 変換によって原ブロックの水平微分画像と垂直微分画像に生じうる基本的関係は、以下の4つである。

- 回転・反転
- 水平微分画像と垂直微分画像の入れ替え
- 水平および垂直微分画像ともに符号反転
- 水平および垂直微分画像どちらか一方のみ符号反転

EtC 変換で生じる微分画像のすべての関係は、これらの4つの組み合わせで表現される。従って、この4つの関係について、HOG 特徴量への影響についてさらに考察する。

### A. 回転・反転

微分画像の回転・反転は、ヒストグラムに基づく HOG 特徴抽出に影響を与えないのは自明である。

### B. 水平微分画像と垂直微分画像の入れ替え

式(3)より、水平微分画像 $l_x$ と垂直微分画像 $l_y$ が入れ替わっても勾配強度 $I(x, y)$ には影響がない。勾配方向 $\theta(x, y)$ の関係は、式(4)より、

$$\begin{aligned} \theta'(x, y) &= \tan^{-1} \frac{l_x(x, y)}{l_y(x, y)} \\ &= \begin{cases} \frac{\pi}{2} - \theta(x, y) & (0 \leq \theta(x, y) < \frac{\pi}{2}) \\ -\frac{\pi}{2} - \theta(x, y) & (-\frac{\pi}{2} < \theta(x, y) < 0) \end{cases} \quad (10) \end{aligned}$$

となる。ここで、 $\theta'$ は EtC 変換後における勾配方向であり、 $\theta, l_x, l_y$ はそれぞれ変換前における勾配方向、水平微分画像、垂直微分画像である。式(10)より、 $l_x$ と $l_y$ の入れ替えによって勾配方向は、 $0 \leq \theta(x, y) < \frac{\pi}{2}$ のときは $\frac{\pi}{4}$ 対称変換され、

$-\frac{\pi}{2} < \theta(x, y) < 0$  のときは  $-\frac{\pi}{4}$  対称変換されることが分かる。この場合、次に生成される勾配方向ヒストグラムでは、ヒストグラムのビンの置換に相当する。ヒストグラムのビンの置換は、画像間で共通の置換操作であれば（鍵が共通）、特徴量ベクトルの単純な位置の入れ替えに相当し、HOG 特徴量としての本質に影響を与えない。よって、水平微分画像と垂直微分画像の入れ替えは、HOG 特徴量抽出に影響しない。

#### C. 水平および垂直微分画像ともに符号反転

式 (3) より、微分画像の符号反転は勾配強度  $I(x, y)$  影響しない。また、勾配方向についても、式 (4) より、

$$\begin{aligned}\theta'(x, y) &= \tan^{-1} \frac{-l_y(x, y)}{-l_x(x, y)} \\ &= \theta(x, y)\end{aligned}$$

となり、勾配方向生成に影響はない。よって、水平および垂直微分画像ともに符号反転されても、HOG 特徴量抽出への影響はない。

#### D. 水平および垂直微分画像どちらか一方のみ符号反転

勾配強度についてはパターン **C** のときと同様に、影響を受けない。また、勾配方向については、式 (4) より、

$$\begin{aligned}\theta'(x, y) &= \tan^{-1} \frac{l_y(x, y)}{l_x(x, y)} \\ &= -\theta(x, y)\end{aligned}$$

となり、勾配方向は符号反転の影響を受ける。そのため、次に生成される勾配方向ヒストグラムは反転変換されることになるが、これは **B** の場合と同様に、HOG 特徴量としての本質に影響を与えるものではない。よって、水平および垂直微分画像どちらか一方のみ符号反転されても、HOG 特徴量抽出への影響はない。

以上の議論によって、EtC 画像の各ブロックから微分画像を生成する場合には、HOG 特徴量抽出には影響がないことが分かる。

### 3.3 画像全体での微分画像

ここまで、EtC 画像のブロック内の微分画像について考察し、同一鍵を用いる場合は HOG 特徴量抽出に影響しないことを述べた。次に、EtC 画像全体から、HOG 特徴量を抽出するために、ブロックごとに微分画像を生成し、それらを連結したものを EtC 画像全体の微分画像とする手法を提案する。

図 5 は、画像全体の微分画像を EtC 画像から取得する手順である。まず、EtC 画像  $\hat{L}$  をサイズ  $B \times B$  のブロックに分割し、 $\hat{L}_t, t = 1, 2, \dots, T$  とする。ただし  $T$  はブロックの総数である。次に、各  $\hat{L}_t$  について、式 (1) および式 (2) に従い水平方向および垂直方向の微分画像  $\hat{l}_{tx}, \hat{l}_{ty}$  をそれぞれ生成する。最後に  $\hat{l}_{tx}, \hat{l}_{ty}$  それぞれについて全ブロック分を連結して、水平方向および垂直方向の微分画像  $\hat{l}_x, \hat{l}_y$  とする。この提案法では、ブロックをまたいだ画素間の差分をとらない。このようにして生成された微分画像を用いて、式 (3) および式 (4) に従い画像全体の  $I, \theta$  を生成し、HOG セルおよび HOG ブロックを定義して HOG 特徴とする。

上記の手法は、EtC 画像のブロック内の微分画像が HOG 特徴抽出に影響がないことを指摘して、EtC 画像全体の微分画像を得るための手法である。しかし、HOG ブロックの定義が、EtC 画像の複数のブロックにまたがる場合、HOG ブロックごとのヒストグラム正規化処理などの影響を少し受け、実効結果が完全に一致するわけではない。

## 4. 実験

提案法の妥当性を、原画像から抽出された HOG 特徴量との比較実験によって評価する。本実験では、SVM, 決定木, 1 分割決定木を弱分類器とする AdaBoost を用いた画像の 2 クラス分類問題を考える。より客観的に特徴抽出性能を評価するため、EtC ブロックサイズと HOG セルサイズを様々に変更し、さらに複数の分類器アルゴリズムを準備して、性能評価を行った。

### 4.1 実験条件

本実験では、顔認証システムの客観評価用データベース FERET [9] [10] を用いた。FERET データベースの顔画像データは、顔の向き、人種、口ひげの有り無し等でアノテーションが与えられている。本実験では、顔の向きが正面であることを表す FERET でのアノテーション fa および fb の画像を「正面顔」クラス（正例）、顔の向きが真左であることを表す pl が付与された画像を「真左顔」クラス（負例）として、HOG 特徴量を用いた 2 クラス画像分類を行う。FERET データベースからは、「正面顔」クラスが 2722 枚、「真左顔」クラスが 1312 枚それぞれ得られ、計 4034 枚の画像が実験用に得られた。これらの画像からランダムに 1009 枚を選び教師画像データとし、残りを評価用のクエリデータとした。

画像サイズは一律で  $256 \times 384$  であり、カラー画像は白黒化された。HOG セルサイズと EtC ブロックサイズは同一の  $B \times B$  とし、 $B = 8, 16, 32$  それぞれの場合に対して実験を行った。各 HOG ブロックは図 2 の例と同じく  $2 \times 2$  の HOG セルを含む。HOG ブロックのシフトは、図 2 のようにセルの重複を許容するように行われる。EtC 変換を行うかどうか、微分画像の生成法の違いによって、以下の 3 つの条件を設定した。

- 条件 A(従来法) : EtC 変換なし、微分画像 (画像全体から直接生成)
- 条件 B(参考) : EtC 変換あり、微分画像 (EtC 画像全体から直接生成)
- 条件 C(提案法) : EtC 変換あり、微分画像 (EtC ブロック単位で生成)

さらに、以上の条件で求めた特徴ベクトルを用いて、線形 SVM, ガウシアンカーネル SVM, 決定木, 1 分割決定木を弱分類器とする AdaBoost の 4 つの分類器を学習した。

### 4.2 実験結果

学習された分類器に評価用クエリデータを適用し、性能評価を行った。評価尺度は、ROC (Receiver Operating Characteristic) 曲線の下側面積である AUC (Area under the curve) を用いた。ROC 曲線は、2 値分類において分類スコアから正例と判断される閾値を操作して作成するグラフであり、縦軸を真陽性

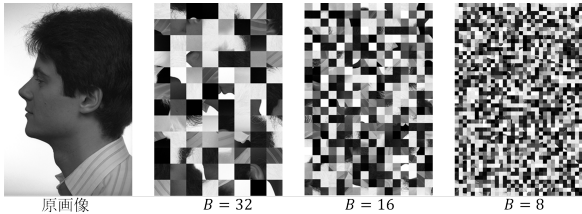


図 6 FERET データベース内の原画像とその EtC 画像例  
表 1 AUC 値 (線形 SVM)

-	条件 A	条件 B	条件 C
$B = 8$	.999897	.999003	.999808
$B = 16$	.999853	.999768	.999795
$B = 32$	.999732	.999637	.999792

表 2 AUC 値 (ガウシアンカーネル SVM)

-	条件 A	条件 B	条件 C
$B = 8$	.999937	.999765	.999840
$B = 16$	.999956	.999764	.999909
$B = 32$	.999980	.999553	.999927

率, 横軸を偽陽性率とする. AUC が大きいほど良い分類器である. SVM であれば式 (6) の識別関数, 決定木であれば分類出力ノードにおける教師データの事後確率, AdaBoost であれば重み付き多数決の値がそれぞれ分類スコアとなる.

以下に分類器ごとに結果の考察を示す. 表 1 および表 2 に線形 SVM, ガウシアンカーネル SVM の AUC 値をそれぞれ示す. これより,  $B = 8, 16$  の線形 SVM とガウシアンカーネル SVM,  $B = 32$  の線形 SVM は条件 A, 条件 C, 条件 B の順で良い分類性能を獲得していることが分かる. しかし,  $B = 32$  の線形 SVM のみ条件 C, 条件 A, 条件 B の順で良い分類性能を獲得している.

次に, 表 3 に決定木の AUC 値を示す. これより,  $B$  の値に関わらず条件 A, 条件 C, 条件 B の順で良い分類性能を獲得していることが分かる.

最後に, 表 4 に 1 分割決定木の AdaBoost の AUC 値を示す. これより,  $B = 8, 16$  の場合は条件 A, 条件 C, 条件 B の順で良い分類性能を獲得し,  $B = 32$  の場合は条件 C, 条件 A, 条件 B の順で良い分類性能を獲得していることが分かる.

以上の結果から, 条件 B に比べて条件 C が優れていることが分かる. これは, EtC 画像からの HOG 特徴抽出においては, ブロック単位で微分画像を求めることが有効であることを表している. しかし, ほとんどの場合条件 C は条件 A と比べて, 分類性能が落ちていることが分かる. これは提案法の, EtC ブロック境界をまたいだ画素間の関係が抽出できないというデメリットによる影響であると考えられる. ただし,  $B = 32$  のときのみ, 線形 SVM および 1 分割決定木の AdaBoost の 2 つの分類器において, 条件 C が条件 A よりも良い分類性能を獲得している. これは,  $B$  が大きいために提案法より上記のデメリットによる影響を受けづらかったからであると考えられる.

## 5. まとめ

本稿では, 各種機械学習法への適用を前提として, EtC 画像からの HOG 特徴抽出法を提案した. まず, EtC 画像の同一ブ

表 3 AUC 値 (決定木)

-	条件 A	条件 B	条件 C
$B = 8$	.881518	.855258	.867289
$B = 16$	.908466	.864560	.865899
$B = 32$	.929177	.867602	.894770

表 4 AUC 値 (1 分割決定木の AdaBoost)

-	条件 A	条件 B	条件 C
$B = 8$	.998654	.996732	.998040
$B = 16$	.999340	.998625	.999186
$B = 32$	.999474	.997981	.999573

ロック内における EtC 変換は, 同一鍵を画像間で用いた場合, HOG 特徴抽出に本質的な影響がないことを理論的に示した. その上で, EtC 画像全体から HOG 特徴量を抽出するために, ブロックごとに微分画像を生成し, それらを連結したものを EtC 画像全体の微分画像とする手法を提案した. 最後に, 抽出された特徴量を用いて各種機械学習法による分類器を学習し, 提案法の妥当性を評価した.

謝辞 Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD Counterdrug Technology Development Program Office.

## 文 献

- [1] C.-T. Huang, L. Huang, Z. Qin, H. Yuan, L. Zhou, V. Varadharajan, and C.-C.J. Kuo, "Survey on securing data storage in the cloud," APSIPA Transactions on Signal and Information Processing, vol.3, no.e7, pp.1-17, 2014.
- [2] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for jpeg/motion jpeg standard," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.98, no.11, pp.2238-2245, 2015.
- [3] T. Chuman, K. Kurihara, and H. Kiya, "Security evaluation for block scrambling-based etc systems against extended jigsaw puzzle solver attacks," Multimedia and Expo (ICME), 2017 IEEE International Conference on IEEE, pp.229-234, 2017.
- [4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on IEEE, pp.886-893, 2005.
- [5] 高村大也, 言語処理のための機械学習入門, コロナ社, 2010.
- [6] L. Breiman, Classification and regression trees, Routledge, 2017.
- [7] Y. Freund and R.E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," Journal of computer and system sciences, vol.55, no.1, pp.119-139, 1997.
- [8] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for jpeg/motion jpeg standard," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.98, no.11, pp.2238-2245, 2015.
- [9] P.J. Phillips, H. Wechsler, J. Huang, and P.J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," Image and vision computing, vol.16, no.5, pp.295-306, 1998.
- [10] P.J. Phillips, H. Moon, S.A. Rizvi, and P.J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.22, no.10, pp.1090-1104, Oct. 2000.