行列分解に基づく事例型識別器の高速化に関する研究

TP14010 **黒川 貴都**　　**指導教授：藤吉弘亘**

1. はじめに

事例ベースの識別器の学習法である Exemplar-SVM(E-SVM)[1] は，多様な見えの変化に対して頑健に識別が可能であり，多くのサンプルを保持することなく，最近傍探索が可能である．E-SVM による識別は，ポジティブサンプルの数だけ識別器を必要とし，その全ての識別器に未知のサンプルを入力する必要がある．そのため，識別時には膨大な処理時間を要する．そこで，本手法では識別器に対し近似計算を導入した高速化手法である，二値ベクトル分解 [2, 3] の考えに基づき E-SVM の高速化を行う．E-SVM の重みベクトルの相関を元にグルーピングし，各グループに対して行列分解を行うことで識別の高速化を実現する．

2. 提案手法

本手法では，E-SVM の高速化のために，識別器の重みベクトル $\mathbf{w}$ をスタックした重み行列 $\mathbf{W}$ を基底数 k の実数スケール係数行列 $\mathbf{C}$ と二値基底行列 $\mathbf{M}$ に分解し，近似内積計算を行う．分解をより効率的に行うために相関の高い重みベクトルのみから重み行列 $\mathbf{W}_g(g = 1, 2, \ldots, G)$ を作成するためのグルーピングを行い，各グループの重み行列 $\mathbf{W}$ に対して実数スケール係数行列 $\mathbf{C}$ と二値基底行列 $\mathbf{M}$ に分解する．

2.1. 行列分解

E-SVM は Exemplar 数 (ポジティブサンプル数)E 個の二クラス識別器を用いて多様な見えの変化に対して頑健な識別器を学習している．図 1 に示すように，Exemplar 数の重みベクトル $\mathbf{w} \in \mathbb{R}^D$ が算出され，これら全ての重みベクトル $\mathbf{w}_i$ $(i = 1, 2, \ldots, E)$ と入力特徴ベクトル $\mathbf{x} \in \{-1, 1\}^D$ との内積計算が必要となるため処理時間を要する．この問題に対し本手法では，図 1(b) に示すように，重みベクトルをスタックした重み行列 $\mathbf{W} \in \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_E\}$ を基底数 k の二値基底行列 $\mathbf{M} \in \{-1, 1\}^{D \times k}$ とスケール係数行列 $\mathbf{C} \in \mathbb{R}^{k \times E}$ に分解する．重み行列の分解は，式 (1) のコスト関数を最小化するように二値基底行列 $\mathbf{M}$ とスケール係数行列 $\mathbf{C}$ を最適化する．

$$||\mathbf{W} - \mathbf{MC}||_F^2 \tag{1}$$

Algorithm 1 行列分解

function matrix_decomposition ($\mathbf{W}$, L, k, $\mathbf{X}$)
for $l = 1$ to L **do**
　$\mathbf{M}_l$ を $\{-1, 1\}$ の乱数で初期化.
　$\mathbf{C}_l$ を実数の乱数で初期化.
　repeat
　　$\mathbf{C} = (\mathbf{M}^{\mathrm{T}}\mathbf{M})^{-1}(\mathbf{M}^{\mathrm{T}}\mathbf{W})$
　　$\mathbf{M} = \underset{\mathbf{M} \in \{-1,1\}^{D \times k}}{\arg\min} \|\mathbf{W} - \mathbf{MC}\|_F^2$
　until 式 (1) が収束
end for
$\{\mathbf{M}_l\}$ と $\{\mathbf{C}_l\}$ のうちコスト関数 E を最小とする二値基底行列と基底スケール係数行列を $\hat{\mathbf{M}}, \hat{\mathbf{C}}$ とする.
return $\hat{\mathbf{M}}$, $\hat{\mathbf{C}}$

二値ベクトル間の内積計算は，ハミング距離の計算に置き換えることができるため，高速な演算が可能となる．しかし，行列分解では，全ての重みベクトルに対して共通の二値基底行列 $\mathbf{M}$ を用いて表現するため，識別器の重みベクトル間に相関がない場合は，分解前の識別精度を維持するために多くの基底数が必要となる．すると，図 1 に示すように重みベクトルを独立で分解するベクトル分解 [2, 3] に比べ計算時間が増加するという問題がある．

2.2. 分解対象行列のグルーピング

本手法では，ベクトル間の距離に基づき相関の高い重みベクトルのグルーピングを行う．ベクトル $\mathbf{w}_i(i = 1, 2, \ldots, E)$

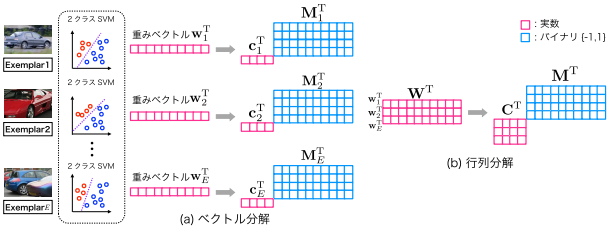

図 1：重み行列 $\mathbf{W}$ の分解

とベクトル $\mathbf{w}_j(j = 1, 2, \ldots, E)$ の距離を式 (2，3) に示す.

$$z = \max\{||\mathbf{w}_a||\ a = i, j\} \tag{2}$$

$$ND = -\left(\frac{||\mathbf{w}_i - \mathbf{w}_j||}{z} - 1\right) \tag{3}$$

距離 ND は $-1 \sim 1$ の実数であり，$ND = 1$ であるとき，重みベクトル $\mathbf{w}_i$ と $\mathbf{w}_j$ が同じとなり，$ND = -1$ であるとき，重みベクトル $\mathbf{w}_i$ と $\mathbf{w}_j$ は反対の向きのベクトルを表す．グルーピングでは，しきい値 th^{corre} と ND を使用し $|ND| > th^{corre}$ であるベクトルを同じグループとする．

2.3. グルーピングを用いたカスケード構造の構築

行列分解は二値基底行列 $\mathbf{M}$ について-1 と 1 の全ての組 み合わせを試行するため，基底を 1 つ増やすと分解に必要な時間が 2 倍となる．そのため，基底数が増加すると現実的に分解できない．そこで，逐次的に行列分解し，カスケード構造を構築することで分解に必要な時間を削減する．また，カスケード構造の識別器を構築することで，識別時に近似内積計算の早期棄却ができるため，演算回数の削減が可能となる．カスケード構造の構築は，1 段目で重み行列 $\mathbf{W}$ に対して行列分解を行い，2 段目以降では残差行列 $\mathbf{R}$ に対して行列分解を行う．残差行列 $\mathbf{R}$ は，前の段で相関がないベクトルも分解により相関が高くなることがある．そこで，図 2 に示すように残差行列においてもグルーピングすることで，各段で最適なグループを作成する．グルーピングと行列分解によるカスケード構造の構築法を **Algorithm2** に示す.

Algorithm 2 カスケード構造の構築

Require: $\mathbf{W}$, k, N, L
　$\mathbf{R} \leftarrow \mathbf{W}$
　for $n = 1$ to N **do**
　　残差行列 $\mathbf{R}$ における重みベクトル間の距離を式 (2, 3) より算出
　　$|ND|$ がしきい値 th^{corre} 以上である重みベクトルをグルーピングし G 個の残差行列 $\mathbf{R}$ を作成
　　for $g = 1$ to G **do**
　　　残差行列 $\mathbf{R}_g$ を行列分解により $\hat{\mathbf{M}}_n$ と $\hat{\mathbf{C}}_n$ に分解
　　end for
　　$\mathbf{R} \leftarrow \mathbf{R} - \hat{\mathbf{M}}_n\hat{\mathbf{C}}_n$
　end for
　return $\{\hat{\mathbf{M}}\}$, $\{\hat{\mathbf{C}}\}$

Algorithm2 では，分解時はそれぞれのグループ毎で基底数 k の行列分解を行う.

2.4. カスケード型識別器による早期棄却

Algorithm2 で構築したカスケード構造を持つ識別器により識別を行う． カスケード型識別器は段数 $n(n = 1, 2, \ldots, N)$ が多いほどより正確に識別できる． しかし，識別境界から大きく離れたネガティブサンプルは少ない段で判定することが可能である．そこで，さらなる高速化のために，識別境界から大きく離れた非検出対象クラスに対して近似内積計算を打ち切るための早期棄却法を導入する．

識別時は，図 3 のように各段の近似内積計算で算出されたスコアとしきい値 $\mathbf{t} \in \mathbb{R}^E$ により非対象クラスの棄却する．

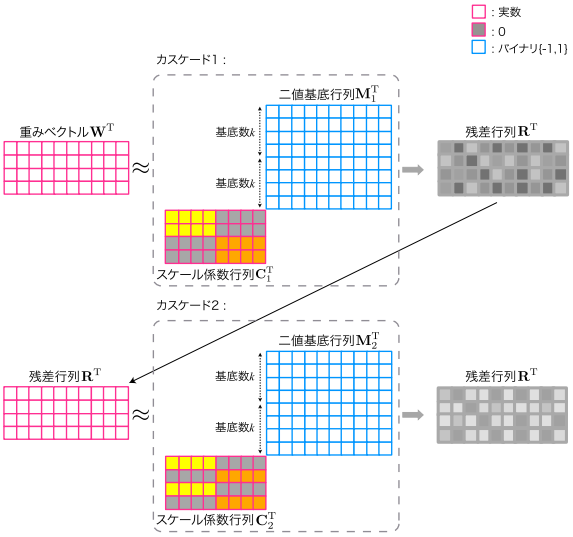

図 2：グルーピングを用いたカスケード構造の構築

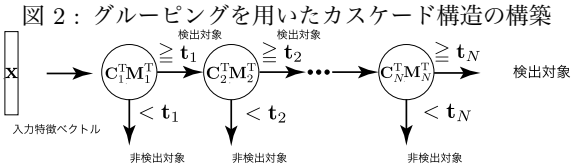

図 3：カスケード型識別器による早期棄却

n 段目における各識別器の出力 $\mathbf{s}_{n,j} \in \mathbb{R}^E$ は式 (4) の近似内積計算により算出する．

$$\mathbf{s}_{n,j} = \sum_{i=1}^{n} \mathbf{C}_i^{\mathrm{T}} \mathbf{M}_i^{\mathrm{T}} \mathbf{x} \tag{4}$$

$\mathbf{M}_i^{\mathrm{T}}\mathbf{x}$ は二値同士の内積となるので，$\mathbf{x}$ の次元数を D としたとき，式 (5) で計算することができる．

$$\mathbf{M}_i^{\mathrm{T}}\mathbf{x} = D - 2\mathrm{HammingDistance}(\mathbf{M}_i, \mathbf{x}) \tag{5}$$

式 (5) の HammingDistance は XOR 演算とビットカウントで計算できるため，高速な演算が可能となる．ビットカウントは，Streaming SIMD Extensions(SSE)4.2 より実装されている POPCNT 関数を使用することで高速な，演算が可能となる．

非対象クラスの棄却は段数 n の近似内積計算により算出されたスコア $\mathbf{s}_n$ をしきい値処理することにより決定する．このとき，カスケードのしきい値 $\mathbf{t}$ はすべて 1 とする．入力画像を I，入力画像から得られる検出ウィンドウの総数を L としたときの近似内積計算と早期棄却による物体検出を **Algorithm**3 に示す．

Algorithm 3 提案手法による物体検出

Require: $\mathbf{M}$, $\mathbf{C}$, I, N
  for $l = 1$ to L **do**
    検出ウィンドウ $I(l)$ から特徴ベクトル $\mathbf{x}_l$ を抽出．
    for $n = 1$ to N **do**　　//N：カスケード数
      $\mathbf{s}_n = \sum_{i=1}^{k_n} \mathbf{C}_i^{\mathrm{T}} \mathbf{M}_i^{\mathrm{T}} \mathbf{x}_l$
      if $\mathbf{s}_n < \mathbf{t}_n$ **then**
        $y_l = 0$
        break //近似計算の打ち切り
      end if
    end for
    $y_l = \arg\max_{j \in J} \mathbf{s}_{N,j}$
  end for

  return $y_1, y_2, \ldots, y_L$

3. 評価実験

実験に用いる識別器は E-SVM とし，データセットの画像から抽出した B-HOG 特徴量 [4] を使用する．また，識別精度と 1 枚あたりの識別時間を比較する． 識別時間の比較では，CPU:Intel Xeon CPU X7542@2.67GHz, RAM:256GB の PC を用いる．評価実験では，Hare らの分解法と Yamauchi らの分解法を適用した Exemplar-SVM の識別器と比較する．実験に用いるデータセットは，PASCAL VOC2007 としクラスは car を使用する．学習時における Exemplar の総数は 253 枚で，ネガティブサンプルは約 500000 枚である．識別時にはポジティブサンプル 553 枚とネガティブサンプル約 700000 枚を使用する．False Positive par window = 0.01 であるときの識別時間と識別精度を図 4 に示す．図 4 より識別精度が同等であるとき，

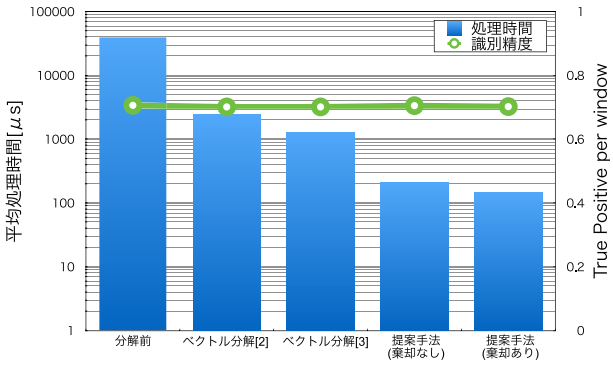

図 4：識別時間と識別精度

棄却無しの提案手法は分解前に比べ約 140 倍，棄却を行うことで約 200 倍高速な識別を可能とした．

4. おわりに

本研究では Exemplar-SVM で学習した識別器に対してグルーピングと行列分解を行うことで，分解前に比べ約 200 倍高速な識別を実現した．今後はその他のオブジェクトに対して提案手法を適用させる．

参考文献

[1] Malisiewicz, T. *et al.*, “Ensemble of exemplar-svms for object detection and beyond, ” ICCV, pp89–96, 2011

[2] Hare, S. *et al*, “Efficient online structured output learning for keypoint-based object tracking, ” CVPR, pp. 1894–1901, 2012

[3] Yamauchi, Y. *et al*, “Asymmetric Feature Representation for Object Recognition in Client Server System, ” ACCV, 2014

[4] Yamauchi, Y. *et al*, “Relational HOG Feature with Wild-Card for Object Detection, ” ICCV, 2011

研究業績

[1] 黒川貴都, 山内悠嗣, 安倍満, 山下隆義, 藤吉弘亘 ”高速な多クラス物体検出のための行列分解と早期棄却”, 第 17 回 画像の認識・理解シンポジウム (MIRU2014), SS3-42, 2014.

[2] Kurokawa, T., Yamauchi, Y., Yamashita, T., Fujiyoshi, H., ”Fast Discrimination by Early Judgment Using Linear Classifier”, IAPR International Conference on Machine Vision Applications(MVA2015), pp. 234-237, 2015.

（他 学会口頭発表 3 件）

受賞

[1] MIRU 若手プログラム 2014 Honorable Mention Wakate Presentation Award