

カーネル部分最小二乗分析を用いた歩行者認識

阿部 巖^{†1} 岡谷 貴之^{†1} 出口 光一郎^{†1}

歩行者認識の手法として, Shwartz らは Histograms of Oriented Gradients (HOG) にテクスチャおよび色情報を表現する特徴量を付加し, 表現力に富んだ高次元な特徴量を用いる手法を提案した. 彼らはその高次元な特徴量を一般的な機械学習の手法を用いて分類するために部分最小二乗分析 (PLS) を用いて次元削減を行い, 良好な結果を得た. 本稿では次元削減手法に着目して, サポートベクトルマシン等でも利用されているカーネル法を PLS に適用したカーネル部分最小二乗分析 (KPLS) を用いることを検討した. INRIA 人画像データセットを用いて, PLS と KPLS の性能の比較実験を行ったところ, KPLS は PLS に対して優位な性能を示した.

Pedestrian Detection Using Kernel Partial Least Squares Analysis

TAKASHI ABE,^{†1} TAKAYUKI OKATANI^{†1}
and KOUICHIRO DEGUCHI^{†1}

Shwartz *et al.* have recently proposed a method for pedestrian detection that uses a very high-dimensional, discriminative feature obtained by combining HOG descriptors with additional color and texture features. To deal with the high dimensional feature by classical machine learning algorithms, they employed Partial Least Squares (PLS) Analysis, an efficient dimensionality reduction technique, and reported promising results. In this paper, focusing on dimensionality reduction, we examine the possibility of applying Kernel Partial Least Squares (KPLS) Analysis, a variant of PLS that uses the kernel method widely used in other classification/recognition methods such as SVM. We experimentally compare PLS and KPLS in terms of detection accuracy using the INRIA pedestrian dataset. The results show that KPLS outperforms PLS.

^{†1} 東北大学
Tohoku University

1. はじめに

歩行者の認識は物体認識の中でも, 服装等の個人の差異や多様な姿勢変化等が原因で非常に難しい問題である. 物体を認識するためには, 画像から認識に有用な特徴量を作成し分類を行う必要がある. 歩行者認識に適した特徴量として, Dalal と Triggs が提案した Histograms of Oriented Gradients (HOG) がある.¹⁾ HOG は局所領域における輝度勾配方向をヒストグラム化したものである. Zhu らはブロックサイズ可変の HOG を用いて, 複数のブロックサイズで HOG 特徴を算出し, それらを組み合わせることで局所的な勾配方向の情報に加えて, 大域的な情報を表現した.²⁾ Shwartz らは, ブロックサイズ可変の HOG に加えてテクスチャを表現する共起行列と, 色情報を表現する Color Frequency の 2 つの特徴量を HOG に付加することにより, 歩行者の認識について良好な結果を得た.³⁾ こうして作成された特徴量は非常に高次元になるが, その次元を削減するための手法として, 彼らは部分最小二乗分析 (PLS) を用いた. PLS は識別に有効な情報を維持しながら, 高次元の特徴量データを低次元空間へ写像する手法である.

本稿では PLS をカーネル法の枠組みに拡張したカーネル部分最小二乗分析 (KPLS) を用いて次元削減を行うことにより, 性能の向上を目指す. カーネル法とは, データを一旦高次元の空間 (特徴空間) に移してから処理を行う手法のことで, これを用いることで線形のモデルで非線形のデータを解析することができる. カーネル法はサポートベクターマシン (SVM) を非線形な問題に対して適用するためにも用いられており, 様々な問題に対して良好な結果が得られている. 画像から抽出された特徴量データは非線形な性質を持っていると考えられるため, 非線形なデータを扱うことができる KPLS を使うことにより, 通常の PLS に対してよい結果を得られるはずである.

2. 特徴量

人間の姿を捉えるのに, HOG で表されるエッジ情報だけでは不十分であると考えられる. そのため, エッジ情報, テクスチャ情報, 色情報を表現する特徴量をそれぞれ用意し, それらを結合することで強力な特徴量を構成する. エッジ情報を表現するブロックサイズ可変の HOG, テクスチャ情報を表現する Local Binary Pattern (LBP), 色情報を表現する Color Frequency を用いる.

2.1 HOG

画像のエッジ情報を表現する特徴量として HOG を用いる. 本稿ではより多くの情報を

取得するため、ブロックサイズ可変の HOG を用いる。識別領域の大きさは 64×128 である。ブロックサイズは $16 \times 16, 16 \times 32, 32 \times 16, 32 \times 32, 32 \times 64, 64 \times 32, 64 \times 64, 64 \times 128$ の 8 種類とし、ブロックのスライド幅は 8 ピクセルとした。よってブロック数の合計は 404 である。また、 2×2 の子領域の 9 方向のピンを結合するので、ブロックあたりの特徴量は 36 次元となる。よって識別領域全体での HOG 特徴は 14544 次元となる。

2.2 LBP

テクスチャ情報を表現する特徴量として LBP⁴⁾ を用いる。LBP は広い範囲で応用されており、特に顔認識で良い結果を示している。LBP は単調な輝度変化に対しては不変であり、また計算量が小さいことから歩行者認識にも有効だと考えられる。Wang らは LBP を HOG に LBP を付加し、歩行者検出に適用して良い結果を得ている⁵⁾

本稿では、ブロックごとに LBP 特徴量を作成し、分類に利用する。

各画素における LBP 値計算の手順を図 1 に示す。注目画素からの距離 r の周辺ピクセル n 個について、注目画素に対する大小を考え、大きいものを 1 に、そうでないものを 0 とする。左上のピクセルから時計回りに並べると、 n ビットのパターンが完成する。さらに、このパターンをそのままではなく、パターン中で 0, 1 が変化している部分の数を注目画素の LBP 値とする。また、パラメータ u を設定し LBP 値に上限を設ける。LBP 値を L とすると、 $L > u$ のときは、LBP 値は non-uniform であるとする。各画素の LBP 値 L をブロックごとに $0, 1, \dots, u$, non-uniform の $u + 2$ ビンのヒストグラムとしてまとめたものを LBP 特徴して用いる。本稿では、各パラメータについて $n = 8, r = 1, u = 4$ と定めた。

ブロックサイズは $8 \times 8, 8 \times 16, 16 \times 8, 16 \times 16, 16 \times 32, 32 \times 16, 32 \times 32, 32 \times 64$ の 8 種類とし、ブロックのスライド幅は 8 ピクセルとした。したがってブロック数は 741 となり、識別領域全体での LBP 特徴は 4446 次元となる。

2.3 Color Frequency

色情報をあらず特徴量として、Color Frequency³⁾ を用いる。衣服は多様な色をしているので、色情報は有用な特徴にはなりにくい、主に人の頭部、顔に支配的な色が現れやすい。この頭部、顔領域における支配的な色を利用できれば、認識精度向上に寄与するはずである。

画素 (x, y) における Color Frequency $C(x, y)$ 値を以下の式で定義する。

$$C(x, y) \equiv \operatorname{argmax}_{c=R,G,B} m_c(x, y)$$

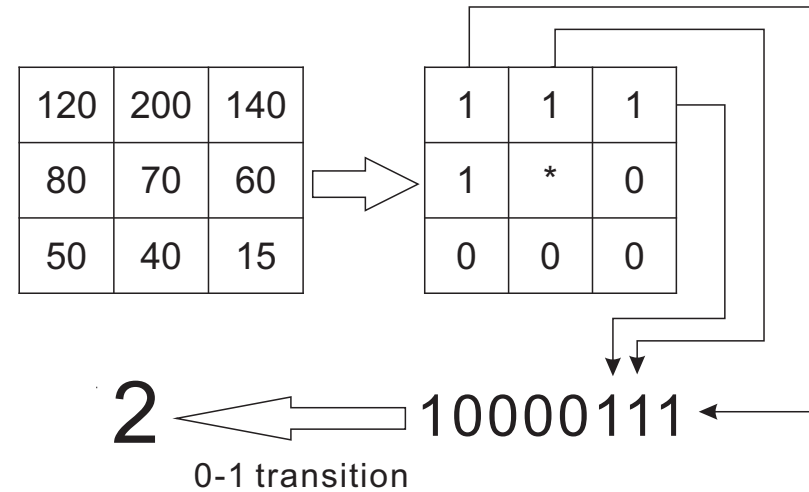


図 1 各画素における LBP 値計算の手順

ここで、 $m_c(x, y)$ は画素 (x, y) における RGB それぞれの勾配強度である。各画素の Color Frequency 値 C をブロックごとに 3 ビンのヒストグラムとしてまとめたものを特徴量として用いる。

ブロックについては HOG と同じものとする。したがってブロック数は 404 となり、識別領域全体での LBP 特徴は 1212 次元となる。

3. カーネル部分最小二乗分析

まず通常の部分最小二乗分析 (以下 PLS) についてまとめ、特徴空間に非線形写像を適用した場合の PLS であるカーネル PLS を述べ、具体的な計算方法を示す。

3.1 部分最小二乗分析 (PLS)

1 つのサンプルの特徴量とそのクラスラベルを、それぞれ N 次元ベクトル $\mathbf{x} = [x_1, x_2, \dots, x_N]^T$ およびスカラー y によって表す。 n 個のサンプル $\{(x_i, y_i)\}$ を考え ($i = 1, \dots, n$)、特徴量をまとめて $n \times N$ 行列 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^T$ 、クラスラベルをまとめて n 次元ベクトル $\mathbf{y} = [y_1, y_2, \dots, y_n]^T$ と書く。このとき $\{\mathbf{x}_i\}$ および $\{y_i\}$ は、サンプル全体の平均がそれぞれ 0、すなわち $\sum_{i=1}^n \mathbf{x}_i / n = 0$ 、 $\sum_{i=1}^n y_i / n = 0$ となるように予め調整しておく。

PLS の核心は、 y がスカラーである今の場合、与えられたサンプル集合に対し、特徴量 x の空間内で x_i と y_i の標本共分散が最大になるような特徴空間の方向 w を求めることにある。そのような w は次のように表される：

$$w = \operatorname{argmax}_{\|r\|=1} [\operatorname{cov}(Xr, y)]^2. \quad (1)$$

ただし $\operatorname{cov}[\cdot]$ は、 n 次元ベクトル a, b に対して $[a, b] = a^\top b/n$ と定義する。このような w を複数個、ただし特徴空間の次元数 N よりもずっと少ない個数選び、それによって特徴量を小さな次元の空間に射影する。サンプルの分布を特徴空間のみで調べて部分空間を選択する主成分分析 (PCA) と異なり、PLS では、クラスラベルとの共分散を考慮して部分空間を選択するため、分類や回帰の性能をずっと向上させられる。

上のような w を 1 つ見つけた後、 $t \equiv Xw$ を計算し、この t に対し $\|X - tp^\top\|^2$ を最小化する p を求める。そして X を $X \leftarrow X - tp^\top$ と修正する。一方の y は、 $\|y - tc\|^2$ を最小化する c によって $y \leftarrow y - tc$ と修正する ($p = X^\top t/\|t\|^2$, $c = y^\top t/\|t\|^2$ となる。) また、後述の説明のために $u \equiv y$ とおく。その後、上の w の計算をもう一度行い、その後 X と y を同様に修正する。以上の計算を p 回繰り返す、 p 個の重みベクトル $W \equiv [w_1, \dots, w_p]$ を得る (X および y の修正方法には幾通りかの方法が存在し、上の方法は文献⁶⁾ の PLS1 に相当する。)

以上の手順は NIPALS (Nonlinear Iterative Partial Least Squares) アルゴリズムとして知られる (これとは別に、固有値問題に帰着して解く方法もある。) t の長さをいつも 1 にすることにして、反復の各回 $k = 1, \dots, p$ に計算される t, p, u をそれぞれ t_k, p_k, u_k と書くと、アルゴリズムは

```
for k = 1 to p do
  w_k = XTy, w_k ← w_k/||w_k||
  t_k = Xw_k, t_k ← t_k/||t_k||
  (u_k = y, u_k ← u_k/||u_k||)
  X ← X - t_k t_kT X
  y ← y - t_k t_kT y
end for
```

となる。

以上の計算によって求まるのは、次のような行列 X および y の分解である：

$$X = TP^\top + F, \quad (2a)$$

$$y = Uq^\top + g. \quad (2b)$$

$T \equiv [t_1, \dots, t_p]$ および $U \equiv [u_1, \dots, u_p]^\top$ は、それぞれ p 個の潜在変数を含む $n \times p$ 行列であり、 $P \equiv [p_1, \dots, p_p]^\top$ および q は、負荷量となる $N \times p$ 行列および p 次元ベクトルである。 F および g は、残差をあらわす $n \times N$ 行列と n 次元ベクトルである。

3.2 PLS による次元削減

このように得た $W = [w_1, \dots, w_p]$ を使って、元の N 次元特徴空間から、それより低い p 次元空間への射影が行える。特徴量 x の、 p 次元空間での像は

$$z = W^\top x \quad (3)$$

のように求められる。

なお、 W は

$$W = X^\top U \quad (4)$$

のようにも与えられ、この関係は後に用いる。これは次のように証明できる⁷⁾。反復 k 回目での修正前の X を X_k , y を y_k と書く。 $w_k = X_k u_k$ であるが、 $X_k = X_{k-1} - t_{k-1} t_{k-1}^\top X_{k-1}$ であるから、 $w_k = X_{k-1}^\top (I - t_{k-1} t_{k-1}^\top) u_k$ と書ける。ここで $l < k$ なる l につき、 $t_l^\top u_k = 0$ である。なぜなら、 $t_k^\top t_l = \delta_{kl}$ より、まず $u_k = y - \sum_{m=1}^{k-1} t_m t_m^\top y$ が示せる。これに t_l^\top を左からかけると、 $t_l^\top u_k = 0$ となると分かる。したがって $w_k = X_{k-1}^\top u_k$ である。 $k-1, k-2, \dots$ とこれをさかのぼると、 $w_k = X_1 u_k$ を得る。 X_1 は元の X と同じであるので、 $W = XU$ となることが分かる。

3.3 カーネル部分最小二乗分析 (Kernel PLS, KPLS)

特徴量 x を、別の S 次元空間 F 上に写す非線形写像 ϕ , すなわち $\phi: x \in \mathbb{R}^N \rightarrow \phi(x) \in F$ を考える。すべてのサンプル $\{x_1, \dots, x_n\}$ を写像 ϕ によってこの空間 F 上に写した上で、 F 上のサンプルについて PLS を適用することとする。これは、元の特徴空間にて元サンプルに対し、非線形回帰を考えることと同じ効果がある。

写像されたサンプルを $\phi_i \equiv \phi(x_i)$ ($i = 1, \dots, n$) と書き、 $n \times S$ 行列 $\Phi \equiv [\phi_1, \dots, \phi_n]^\top$ を定義する。 F 上での $\{\phi_1, \dots, \phi_n\}$ に対する NIPALS アルゴリズムを、 $w_k = X^\top y$ の計算ステップを w_k を $t_k = X w_k$ に代入することで省略し、 X の修正を等価な XX^\top に対す

る修正の形に書き直すと、次のようになる：

```
for k = 1 to p do
  t_k = ΦΦTy, t_k ← t_k/||t_k||
  (u_k = y, u_k ← u_k/||u_k||)
  ΦΦT ← (Φ - t_k t_kTΦ)(Φ - t_k t_kTΦ)T
  y ← y - t_k t_kTy
end for
```

上のアルゴリズムでは $\phi(\mathbf{x})$ の計算が必要だが、これにカーネルトリックを適用すると、これが不要となる。 $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j) = \phi_i^T \phi_j$ とし、 $K(\mathbf{x}_i, \mathbf{x}_j)$ を (i, j) 要素とする $n \times n$ 行列 \mathbf{K} を定義する。 $\mathbf{K} = \Phi\Phi^T$ であり、これを上のアルゴリズムに代入すると

```
for k = 1 to p do
  t_k = Ky, t_k ← t_k/||t_k||
  (u_k = y, u_k ← u_k/||u_k||)
  K ← (I - t_k t_kT)K(I - t_k t_kT)
  y ← y - t_k t_kTy
end for
```

となる。ただし \mathbf{I} は $n \times n$ 単位行列である。

3.4 KPLS による次元削減

(線形) PLS の場合、特徴空間から低次元 (p) 空間への射影は (3) 式によって行った。 \mathbf{W} が (4) 式のように与えられることから、写像 ϕ 適用後の空間における上述の PLS にこれを当てはめると

$$\mathbf{W} = \Phi^T \mathbf{U} \quad (5)$$

となる。したがって、テストサンプルの特徴量 \mathbf{x}_t は $\phi_t \equiv \phi(\mathbf{x}_t)$ として、

$$\mathbf{z} = \mathbf{U}^T \Phi \phi_t \quad (6)$$

のように射影される。カーネルトリック $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi_i^T \phi_j$ により、 $\mathbf{k}_t \equiv [K(\mathbf{x}_1, \mathbf{x}_t), \dots, K(\mathbf{x}_n, \mathbf{x}_t)]^T$ を定義すると

$$\mathbf{z}_t = \mathbf{U}^T \mathbf{k}_t \quad (7)$$

と書き直すことができる。なお文献⁸⁾では、次の射影

$$\mathbf{z}_t = (\mathbf{U}^T \mathbf{K} \mathbf{T})^{-1} \mathbf{U}^T \mathbf{k}_t \quad (8)$$

の使用が記載されているが、違いは射影後の p 次元空間内での線形変換の有無であり、分類をする上ではどちらも同じである (以下の実験では後者を使った)。

4. 実験

PLS と KPLS でそれぞれ次元削減を行い、分類の精度を比較する。

実験のためのデータセットとして、INRIA 人画像データセットを用いる。INRIA 人画像データセットは、学習サンプルとして 2416 枚の人画像と 1218 枚の人ではない画像を、テストサンプルとして 1126 枚の人画像と 453 枚の人ではない画像を含んでいる。

特徴量として HOG, LBP と Color Frequency を結合した 20202 次元の特徴量を用いる。

次元削減後の分類には最小二乗法により学習した線形分類器を用いる。線形 SVM についても検討したが、単純な線形分類器とほとんど性能差がなかったため、線形分類器を採用した。

KPLS で使用するカーネル関数にはガウシアンカーネルを用いた。ガウシアンカーネルは下式で表される。

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\beta|\mathbf{x} - \mathbf{x}'|^2) \quad (9)$$

PLS, KPLS の各種パラメータは実験的に決定した。

4.1 各種パラメータの決定

PLS については削減後の次元数を、KPLS については削減後の次元数とカーネル関数のパラメータを決定する必要がある。最適なパラメータを決定するためにパラメータを段階的に変化させて実験を行い、もっともよい結果であったものを用いる。実験結果は横軸に誤検出率 $\frac{FalsePos}{FalsePos+TrueNeg}$ 、縦軸に見逃し率 $\frac{FalseNeg}{FalseNeg+TruePos}$ をとった DET (Detection Error Trade off) 曲線により示す。

PLS の削減後の次元数 p は、20, 24, 28 と変化させ、分類精度を比較して最も良いものに定める。各次元数における分類精度を図 2 に示す。この結果から $p = 24$ と決定する。

KPLS のパラメータは削減後の次元数 q およびカーネル関数のパラメータ β を段階的に変化させて、分類精度を比較し最も良いものに定める。各条件における分類精度を図 3 に示す。この結果から $q = 15, \beta = 0.003$ と決定する。

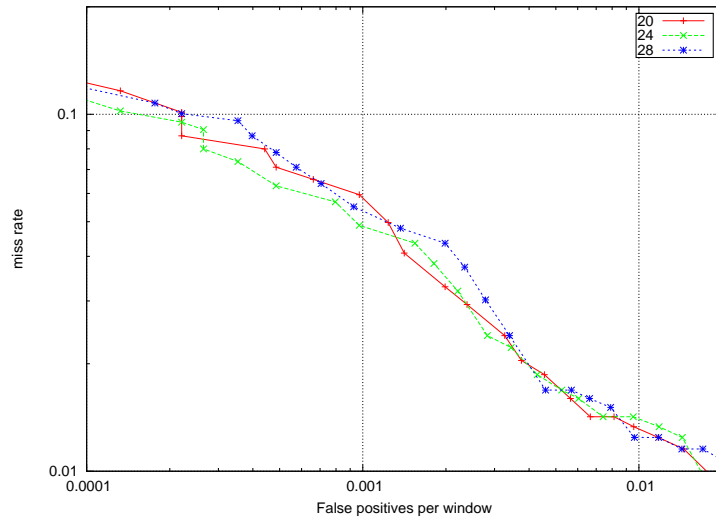


図 2 PLS による削減後の次元数 p に対する DET 曲線

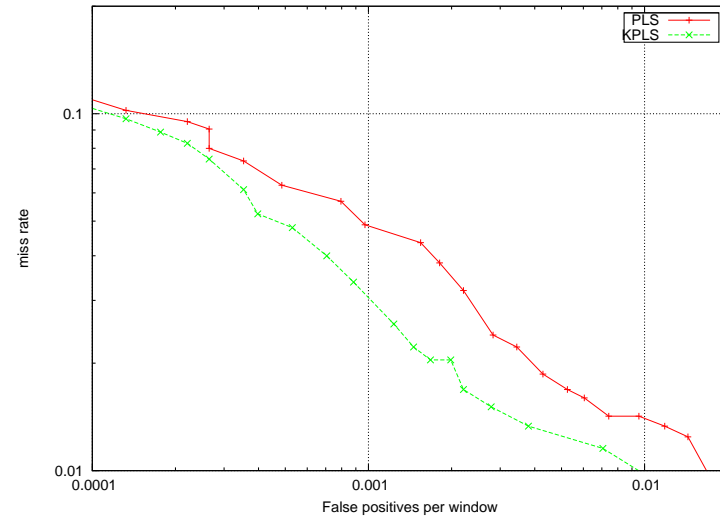


図 4 KPLS, PLS の DET 曲線

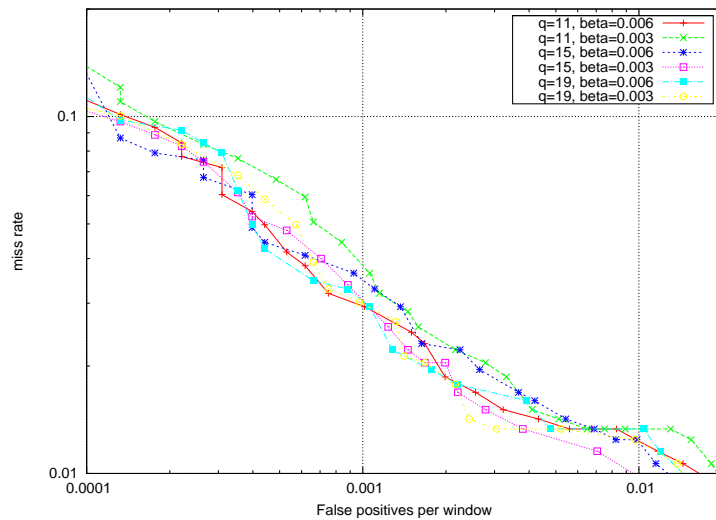


図 3 KPLS による削減後の次元数 q およびカーネル関数のパラメータ β に対する DET 曲線

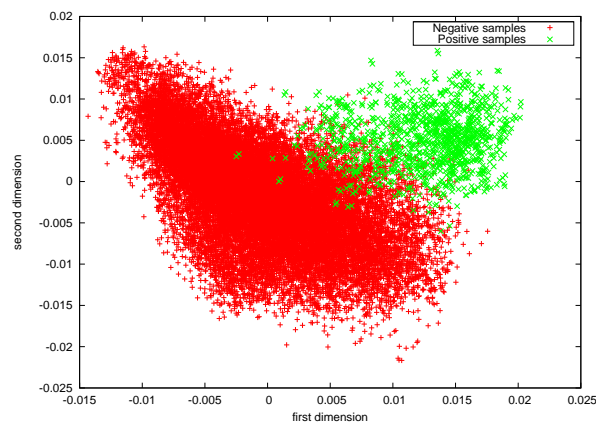
4.2 KPLS vs PLS

図 4 は KPLS と PLS の検出結果を比較したグラフである．KPLS は PLS に対して完全に優位な性能を示していることがわかる．誤検出率 0.1% においては，PLS に比べて見逃し率で 4.3% が 3.0% に向上している．

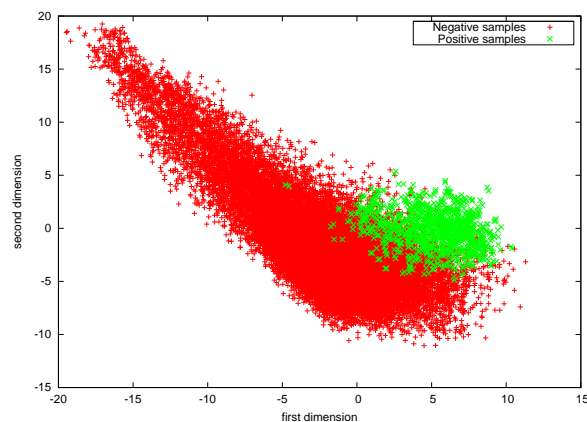
図 5(a) および図 5(b) は，それぞれ KPLS と PLS を用いてテストデータを次元削減後に，第一，第二次元をプロットしたものである．たった 2 次元分のプロットなので正確な評価はできないが，これらの図からも KPLS による次元削減の結果が PLS によるものに対して，分類し易いものになっていることが伺える．

4.3 計算時間の問題

このように，KPLS は分類精度において PLS より優れている．しかし，計算時間の点では KPLS は PLS より劣っている．具体的には，CPU に Corei7-920，メモリに 24GB DDR3 を搭載したコンピュータで学習に約 2600 秒，テストデータの写像に約 400 秒を要した．PLS は学習に約 180 秒，テストデータの写像に約 0.6 秒を要した．学習に要する時間は少々長くても実用上は問題ないが，テストデータの低次元への写像に時間がかかりすぎるのは問題で，たとえば実時間で認識を行いたい際にフレームレートに直接影響する．PLS は写像の際に



(a) KPLS



(b) PLS

図5 KPLS, PLSによる次元削減後のテストデータの第一, 第二次元

単なる行列積を計算するだけであるが, KPLS はカーネル関数の計算を何度も行う必要があるため, 所要時間に大きな違いがある.

5. 結 論

次元数が大きい特徴量の次元削減の手段として, KPLS を用いた. 実験の結果, 他のカーネル法を用いた手法と同様に, KPLS は PLS に比べて, 歩行者認識に対して優位な性能を持つことがわかった.

今後, カーネル SVM と PLS, 線形 SVM と KPLS のように, 分類器の種類と次元削減の方法の組み合わせが性能にどのような影響を与えるかを検討したい.

計算時間の問題については前節で述べたように, 現状では実際のアプリケーションに用いることができる速度は達成できていないが, アルゴリズムの見直しや GPU 上での実装などにより, 識別性能と計算量のバランスをはかれると考えている.

参 考 文 献

- 1) Dalal, N. and Triggs, B.: Histograms of Oriented Gradients for Human Detection, *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol.1, pp.886–893 (2005).
- 2) Zhu, Q., Yeh, M.-C., Cheng, K.-T. and Avidan, S.: Fast Human Detection Using a Cascade of Histograms of Oriented Gradients, *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, IEEE Computer Society, pp.1491–1498 (2006).
- 3) Schwartz, W., Kembhavi, A., Harwood, D. and Davis, L.: Human Detection Using Partial Least Squares Analysis, *Accepted to be presented in the International Conference on Computer Vision* (2009).
- 4) Ojala, T., Pietikainen, M. and Harwood, D.: A Comparative Study of Texture Measures with Classification Based on Feature Distributions, *PR*, Vol.29, No.1, pp. 51–59 (1996).
- 5) Xiaoyu Wang, Tony X.Han, S.Y.: An HOG-LBP Human Detector with Partial Occlusion Handling, *Accepted to be presented in the International Conference on Computer Vision* (2009).
- 6) Rosipal, R. and Krämer, N.: Overview and Recent Advances in Partial Least Squares, *Subspace, Latent Structure and Feature Selection Techniques* (et al., C.S., ed.), Springer-Verlag, pp.34–51 (2006).
- 7) Rännar, S., Lindgren, F., Geladi, P. and Wold, S.: A PLS kernel algorithm for data sets with many variables and fewer objects. Part 1: Theory and algorithm,

Chemometrics and Intelligent Laboratory Systems, Vol.8, pp.111–125 (1994).

- 8) Rosipal, R. and Trejo, L.J.: Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space, *Journal of Machine Learning Research*, Vol.2, pp.97–123 (2001).