

# 異なる部分空間における PCA-SIFT を用いた交通道路標識認識

## Road Sign Detection by PCA-SIFT in Different Feature Subspaces

伊原有仁† 藤吉弘亘† 高木雅成‡ 公文宏明‡ 玉津幸政‡

Arihito Ihara†, Hironobu Fujiyoshi†, Masanari Takagi‡, Hiroaki Kumon‡, Yukimasa Tamatsu‡

† 中部大学, ‡ 株式会社デンソー

† Chubu University, ‡ DENSO CORPORATION

E-mail: †ai@vision.cs.chubu.ac.jp, †hf@cs.chubu.ac.jp

### Abstract

ドライバへの運転支援として、車載カメラで撮影した映像から交通道路標識を認識する手法が既に提案されている。SIFT 特徴量を用いた交通道路標識認識では、SIFT 特徴量を用いているため標識画像の拡大縮小、傾き、オクルージョン、照明変化に対してロバストな認識を実現している。しかし、SIFT 特徴量の抽出とマッチングの処理コストが高いためリアルタイム処理が困難である。さらに、マッチングの際の誤対応が多いため標識以外を誤検出する問題が挙げられている。そこで、本研究では PCA-SIFT を応用し、対象とする標識画像のみを用いた部分空間と、それ以外の一般画像を用いた部分空間を作成し、異なる部分空間における PCA-SIFT 特徴量を抽出する。抽出した特徴量を用いて Real AdaBoost 識別器を構築し、予め背景上の特徴点を削除することで、対応点マッチングの高速化と高精度化を行い、処理時間の短縮、認識率の向上を図る手法を提案する。評価実験の結果、SIFT 特徴量、PCA-SIFT 特徴量を用いた識別器に比べ、提案手法が最も識別性能が高い結果となった。さらに、マッチング候補点を提案手法により選別できたため、従来法に比べ、マッチング処理時間を約 2 分の 1 に短縮することができた。

### 1 はじめに

近年の車社会において、車同士もしくは車と人の交通事故が深刻な問題となっている。その原因の 1 つとして、ドライバによる交通道路標識（以下、道路標識と表記）の見落としが挙げられている。道路標識は、ドライバに対して安全且つ円滑な走行を促すための指示や警告を行っており、ドライバは道路標識に従って走行することで、事故の防止や交通の妨げといった危険を回避することができる。SIFT 特徴量を用いた交通道路標識認識は、SIFT 特徴量を用いているため、道路標識の劣化やオクルージョンの発生、さらにはドライバの道路標識の見落としや誤った対処により、必ずしも守

られるものではない。これらの問題を解決するために、車載カメラの映像に画像処理を施すことで道路標識を自動で認識し、得られた情報をドライバへ提供する研究が行われている [1][2][3]。

従来法には、実際の道路標識画像を用いて識別器を作成し認識を行う手法や、標識の色情報により判別する手法が提案されている。テンプレートマッチングを用いる手法 [1] では、道路標識の大きさの違いや、傾き、照明変化など、あらゆる場面を想定したテンプレートが必要となるため、膨大な処理となる。また、色情報を用いる手法 [2] では、道路標識が実環境下に存在しているため、太陽光の照明変動により様々な色に変化する。そのため、このようなあらゆる照明変動を考慮した道路標識の色抽出は困難である場合が多い。一方、標識のサイズや回転変化にロバストな手法として、高木らは SIFT 特徴量を用いた交通道路標識認識手法 [3] を提案している。この手法は、画像の拡大縮小、傾き、オクルージョン、照明変化に対してロバストな特徴点の検出、特徴量の記述を行う Scale-Invariant Feature Transform(SIFT)[4] 特徴を利用しているため、高精度な認識を実現している。しかし、SIFT 特徴量の抽出には時間を要し、さらに対応点マッチングでは検出された全キーポイントに対して全探索のマッチング処理を行うため計算時間が多いという問題がある。従って、リアルタイム処理が困難であり、さらに道路標識以外への誤対応を含むため Voting 処理に悪影響を及ぼす可能性が考えられる。

そこで、本研究では SIFT を拡張した PCA-SIFT を応用し、キーポイントの持つ特徴量を標識画像クラスの特徴量と、一般画像クラスの特徴量に分け、それぞれを合成した特徴量を用いて Real AdaBoost により識別器の構築を行う。識別器によって背景上と判断されたキーポイントを削除することで対応点マッチングの高速化、高精度化の実現を図り、標識認識における認識率の向上とマッチング処理時間の短縮を図る手法を提案する。

## 2 PCA-SIFT[5]

Keらにより提案されたPCA-SIFTは、SIFTにより検出したキーポイント周辺の勾配情報に対して、主成分分析(PCA)を行う。抽出した勾配情報をPCAにより得られた射影行列を用いて部分空間へ射影し、特徴量次元数を削減する。SIFTは128次元の特徴量をPCA-SIFTでは36次元に圧縮するため、マッチング処理が高速になるという利点がある。以下にPCA-SIFTによる特徴量算出法を示す。

### 2.1 SIFTによるキーポイント検出

PCA-SIFTにおけるキーポイント検出はSIFTと同様の処理となる。SIFTのキーポイント検出は、以下のアルゴリズムで行われる。

- 1 極値探索によるキーポイント検出
- 2 キーポイントのローカライズ
- 3 オリエンテーションの算出

#### 2.1.1 極値探索によるキーポイント検出

キーポイント検出にはスケール探索を用いる。特徴点のスケール探索には、ガウス関数があることがKoenderink[6]やLindeberg[7]によって報告されている。さらに効率的な極値探索法としてDifference-of-Gaussian(DoG)を用いる手法が、Lowe[8]によって提案されている。DoG処理とは、スケールの異なるガウス関数 $G(x, y, \sigma)$ と入力画像 $I(u, v)$ を畳み込んだ平滑化画像 $L(u, v, \sigma)$ の差分をとることである。平滑化画像を以下の式で得る。

$$L(u, v, \sigma) = G(x, y, \sigma) * I(u, v) \quad (1)$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (2)$$

DoGの結果の画像を $D(u, v, \sigma)$ とすると、DoG画像は次式で求まる。

$$D(u, v, \sigma) = L(u, v, k\sigma) - L(u, v, \sigma) \quad (3)$$

この処理を複数のスケール間で行い、複数のDoG画像を得る。得られたDoG画像から極値探索を行う。注目画素に対して隣接する位置と上下スケールの画像領域の画素を比較し、極値であった場合その注目画素の座標をキーポイント候補点として検出する。この検出したキーポイント候補点は検出した画像でのスケールを保持している。

#### 2.1.2 キーポイントのローカライズ

検出したキーポイント候補点には開口問題やDoG値の小さいキーポイントを含んでいる。このようなキー

ポイントを削除しキーポイントとして有効な点のみを残す。まず、キーポイント候補点における2次元ヘッセ行列 $\mathbf{H}$ を次式により計算する。

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (4)$$

$D_{xx}, D_{yy}, D_{xy}$ はそれぞれキーポイント候補位置でのDoG出力値の2次微分から得られる。ここで、ヘッセ行列から求められる第1固有値を $\alpha$ 、第2固有値を $\beta$ ( $\alpha > \beta$ )とする。このときヘッセ行列の対角成分の和 $Tr(\mathbf{H})$ と行列式 $Det(\mathbf{H})$ は次のように計算できる。

$$Tr(\mathbf{H}) = D_{xx} + D_{yy} = \alpha + \beta \quad (5)$$

$$Det(\mathbf{H}) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta \quad (6)$$

さらに、 $\gamma$ を第1固有値と第2固有値の比率とし、 $\alpha = \gamma\beta$ とすると次式のようになる。

$$\begin{aligned} \frac{Tr(\mathbf{H})^2}{Det(\mathbf{H})} &= \frac{(\alpha + \beta)^2}{\alpha\beta} \\ &= \frac{(\gamma\beta + \beta)^2}{\gamma\beta^2} \\ &= \frac{(\gamma + 1)^2}{\gamma} \end{aligned} \quad (7)$$

この値は、固有値 $\alpha, \beta$ の比率で決まる。従って、固有値を求めずにエッジ上の点であるか判別することが可能となる。この値を次式に示すように閾値処理を行うことで、不要なキーポイントを削除する。

$$\frac{Tr(\mathbf{H})^2}{Det(\mathbf{H})} < \frac{(\gamma + 1)^2}{\gamma_{th}} \quad (8)$$

#### 2.1.3 オリエンテーションの算出

ローカライズしたキーポイントのオリエンテーションを求める。オリエンテーションはキーポイントにおける主要勾配方向を表し、特徴量記述の際にオリエンテーション方向に正規化を行うことで、回転に頑健となる。キーポイントのオリエンテーションを求めるには、まずキーポイントが検出された平滑化画像 $L(u, v, \sigma)$ の勾配強度 $m(u, v)$ と勾配方向 $\theta(u, v)$ を以下の式により求める。

$$m(u, v) = \sqrt{f_u(u, v)^2 + f_v(u, v)^2} \quad (9)$$

$$\theta(u, v) = \tan^{-1} \frac{f_v(u, v)}{f_u(u, v)} \quad (10)$$

$$f_u(u, v) = L(u + 1, v, \sigma) - L(u - 1, v, \sigma) \quad (11)$$

$$f_v(u, v) = L(u, v + 1, \sigma) - L(u, v - 1, \sigma) \quad (12)$$

局所領域における勾配強度 $m(x, y)$ と勾配方向 $\theta(x, y)$ から、重み付き方向ヒストグラムを作成する。作成したヒストグラムから最大値となる方向を特徴点の輝度勾配方向として割り当てる。以上の処理により、座標、スケール、オリエンテーションの値を持つキーポイントを検出することができる。

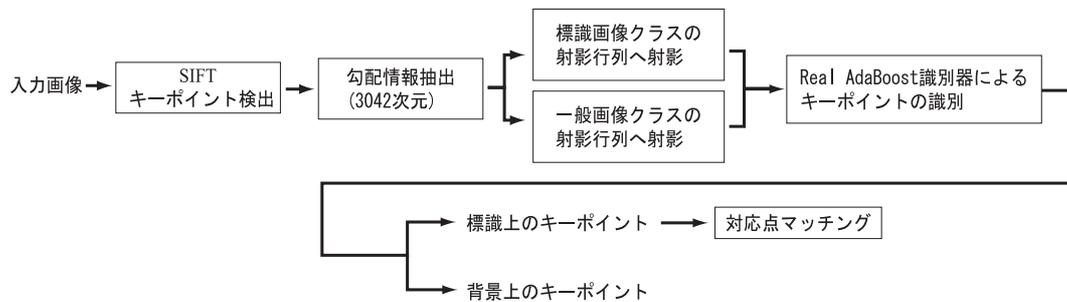


図1 キーポイントの識別

## 2.2 PCA-SIFT による特徴量記述

検出したキーポイントを中心として、スケールを考慮した局所領域をオリエンテーション方向に正規化し、 $41 \times 41$  pixel の patch 領域から水平方向、垂直方向の勾配を算出する (図2)。

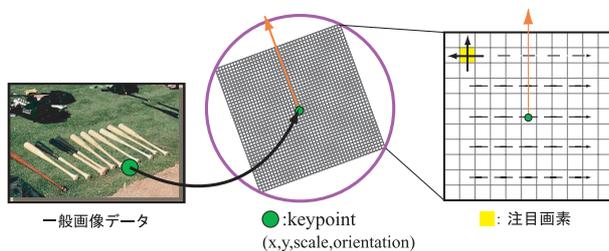


図2 patch からの勾配抽出

これにより、1つのキーポイントに対して3042次元 ( $39 \times 39 \times 2$ ) の勾配特徴量  $x$  を抽出することができる。以上の操作を、大量の一般画像データに対して行い、複数のキーポイントからPCAを用いて部分空間を作成する (図3)。

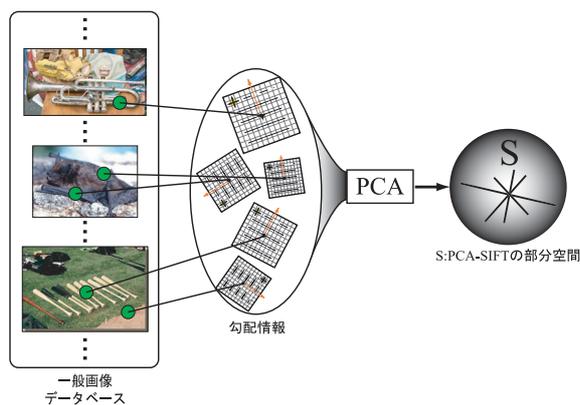


図3 部分空間の作成

PCAにより得られた一般画像上のキーポイントに対する部分空間に、入力画像から同様にして抽出した勾配特徴量を射影しPCA-SIFT特徴量を抽出する。このように得られたPCA-SIFTの特徴量は、SIFTと比べマッチング精度が良く、高速なマッチングが可能というメリットが確認されている。

## 3 提案手法

PCA-SIFTでは、特徴量抽出に用いる部分空間を一般画像データから作成しているのに対して、提案手法では、一般画像上のキーポイントからだけではなく、標識画像上のキーポイントに対する部分空間も作成する。作成した異なる部分空間に射影した特徴量を用いて、キーポイントを識別するためのReal AdaBoost識別器を構築し、入力画像から検出したキーポイントを識別する。提案手法の流れを図1に示す。キーポイントが標識上のキーポイントであると識別されれば、そのキーポイントに対して対応点マッチングを行う。これにより、全探索の対応点マッチングの回避と、背景への誤対応の抑制を行う。以下に処理手順を示す。

- 1 入力画像からSIFTによるキーポイント検出
- 2 キーポイント周辺の水平、垂直勾配特徴抽出
- 3 予め作成しておいた標識画像クラスの部分空間と一般画像クラスの部分空間への射影
- 4 射影した特徴量の合成
- 5 合成した特徴量を用いて、予め構築しておいたReal AdaBoost識別器によるキーポイントの識別
- 6 標識上と判断したキーポイントのみを用いた対応点マッチング

### 3.1 異なる部分空間の作成

提案手法は特徴量を抽出する際に、部分空間を対象物体の画像データベースと対象物体以外の画像データベースから作成する (図4)。一般画像上のキーポイントに対する部分空間では、あらゆる成分を表現する能力を持っているが、局所的な空間の表現能力は低い。そこで、対象とする標識画像上のキーポイントに対する部分空間を作成し、一般画像上のキーポイントに対する部分空間では表現できない局所的な空間を表現する。そのため、一般画像上のキーポイントに対する部分空間では、比較しにくい特徴量が、標識画像上のキーポイントに対する部分空間では表現しやすい特徴量となる。

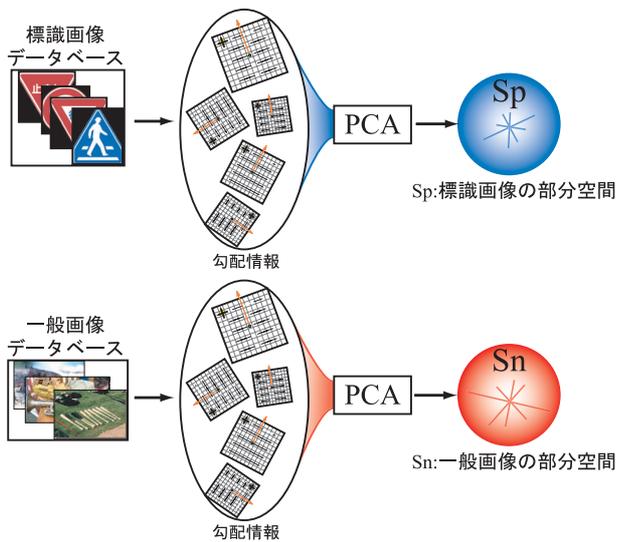


図4 提案手法の部分空間作成

各射影行列の作成に用いる画像を図5に示す。道路標識を標識画像クラス(対象物体)とし、それ以外の物体や背景を一般画像クラス(対象物体以外)とする。標識画像クラスの部分空間は、標識のイラストを様々な変化した画像901枚から抽出し選択したキーポイント77,000点のPCA-SIFT特徴量、一般画像クラスの部分空間は Caltech256[9]から選んだ画像1,277枚から抽出し選択したキーポイント80,000点のPCA-SIFT特徴量を用いる。本手法で使用する特徴量次元数は、累積寄与率が60%以上である72次元を使用する。



図5 部分空間の作成に用いた使用画像データ例

### 3.2 異なる部分空間に射影した特徴量の抽出

作成した一般画像クラスの部分空間と標識画像クラスの部分空間に対して、入力画像から検出したキーポイントの勾配情報を式(13)により射影し、一般画像クラスの特徴量と標識画像クラスの特徴量の合成を行う。

$$V = [x^T S_n, x^T S_p] \quad (13)$$

ここで、 $V$ は合成後の特徴量、 $x$ は水平、垂直勾配、 $S$ は射影行列を表し、 $S_n$ は一般画像クラスの射影行列、 $S_p$ は標識画像クラスの射影行列である。これにより、1

つのキーポイントから一般画像クラスの特徴量と標識画像クラスの特徴量を抽出する。提案手法では、1キーポイントに対して一般画像クラスから72次元の特徴量、標識画像クラスから72次元の特徴量を抽出する。そのため、1つのキーポイントは144次元の特徴量を持つ。抽出した特徴量を用いて、入力画像から検出したキーポイントを Real AdaBoost 識別器により識別する。

### 3.3 Real AdaBoost による識別器の構築

キーポイントの識別には Real AdaBoost[10]を用いる。これにより、識別に使用する弱識別器の数を減らし、識別時間の短縮を行うことが可能となる。図6に、Real AdaBoostによる識別器構築方法について述べる。Real AdaBoostは、学習するポジティブクラスの特徴量とネガティブクラスの特徴量の各次元の確率密度分布から類似度を求め、最も類似していない特徴量の次元数を学習する。

このとき類似度を評価値とするため、実数での学習と識別が可能である。ポジティブクラスとして車載カメラで撮影した画像から切り出した標識画像566枚のキーポイント22,449点、ネガティブクラスとして同じ車載カメラから撮影した標識を含まない画像14枚のキーポイント35,827点を使用する。識別に使用する弱識別器の数は識別時間と識別精度を考慮し100個とした。

### 3.4 キーポイントの識別方法

構築した Real AdaBoost 識別器を用いてキーポイントの識別を行う。まず、入力画像に対して SIFT によるキーポイント検出をする。次にキーポイント周辺の勾配を一般画像クラスの部分空間と標識画像クラスの部分空間へ射影し、抽出した特徴量を合成する。その特徴量を入力として Real AdaBoost 識別器によりキーポイントの識別を行う。標識上と識別されたキーポイントにのみ対応点マッチングを行い、背景上と識別されたキーポイントに対しては処理を行わない。

キーポイントの識別例を図7に示す。赤い点が SIFT によって検出したキーポイント、青い点が標識上に存在すると識別したキーポイントである。図7(a)の入力画像から3,542点のキーポイントが検出された(図7(b))。提案手法は、検出したキーポイントに対して識別を行い、2,150点のキーポイントを削除することができた(図7(c))。図7(c)の標識部分を拡大した画像(図7(e))から、標識上の点を識別できていることがわかる。従って、残り1,392点に対して対応点マッチングを行う(図7(d))。これにより従来法[3]では、全探索していた対応点マッチング時間を大幅に減少させることができる。

### 3.5 対応点マッチング

認識対象とする標識画像上のキーポイントと3.4で Real AdaBoost 識別器によって標識上に存在すると識別

Real AdaBoost のアルゴリズム

1. 前処理

$N$  個の学習サンプル  $x_1, x_2, \dots, x_N$   
 クラスに対応したラベル  $y \{+1, -1\}$  の用意

2. 学習サンプルの重み  $D_t$  を初期化

$$D_{1n} = 1 / N$$

3. For  $t = 1, 2, \dots, T$   $T$  回の学習回数

For  $m = 1, 2, \dots, M$   $M$  個の弱識別器

3.1 弱識別器  $h_t$  の確率密度分布  $W_t$  の作成

$$W_+^j = \sum_{i: j \in J_i^y = +1} D_t(i)$$

$$W_-^j = \sum_{i: j \in J_i^y = -1} D_t(i)$$

3.2 弱識別器  $h(x)$

$$h(x) = \frac{1}{2} \ln \frac{W_+^j + \epsilon}{W_-^j + \epsilon}$$

3.3 弱識別器選択のための評価値  $Z_m$  の算出

$$Z_m = 2 \sum_j \sqrt{W_+^j W_-^j}$$

4. 最も  $Z_m$  が小さい弱識別器を選択

$$h_t = \arg \min Z_{t,m}$$

5. 学習サンプルの重み  $D_t(i)$  の更新

$$D_{t+1}(i) = D_t(i) \exp[-y_i h_t(x_i)]$$

6. 学習サンプルの重み  $D_t(i)$  の正規化

7. 最終識別器

$$H(x) = \text{sign} \left( \sum_{t=1}^T h_t(x) \right)$$

図 6 Real AdaBoost のアルゴリズム

されたキーポイントが保持する特徴量  $V$  を用いてユークリッド距離による対応点マッチングを行う。ユークリッド距離は式 (14) により求められる。このとき提案手法で用いる特徴量は、標識画像上のキーポイントに対する部分空間に射影した特徴量 (72 次元) を使用する。

$$d(V_{I1}, V_{I2}) = \|x_{I1}^T S_p - x_{I2}^T S_p\|^2 \quad (14)$$

$V_{I1}, V_{I2}$  をそれぞれ対象標識画像  $I1$  のキーポイント、未知入力画像  $I2$  のキーポイントとする。求められた  $d(V_{I1}, V_{I2})$  が小さければ、類似度が高いことになる。さらに閾値処理により、類似度が信頼できる値でなければ対応を取らない。

4 キーポイントの識別とマッチング精度の評価

4.1 キーポイントの識別実験

部分空間を対象物体とそれ以外の空間に分ける有効性を示すため、SIFT 特徴量 (128 次元), PCA-SIFT 特徴量 (72 次元), 提案手法の特徴量 (144 次元) をそれぞれ

● : 検出したキーポイント ● : 標識上に存在すると識別したキーポイント

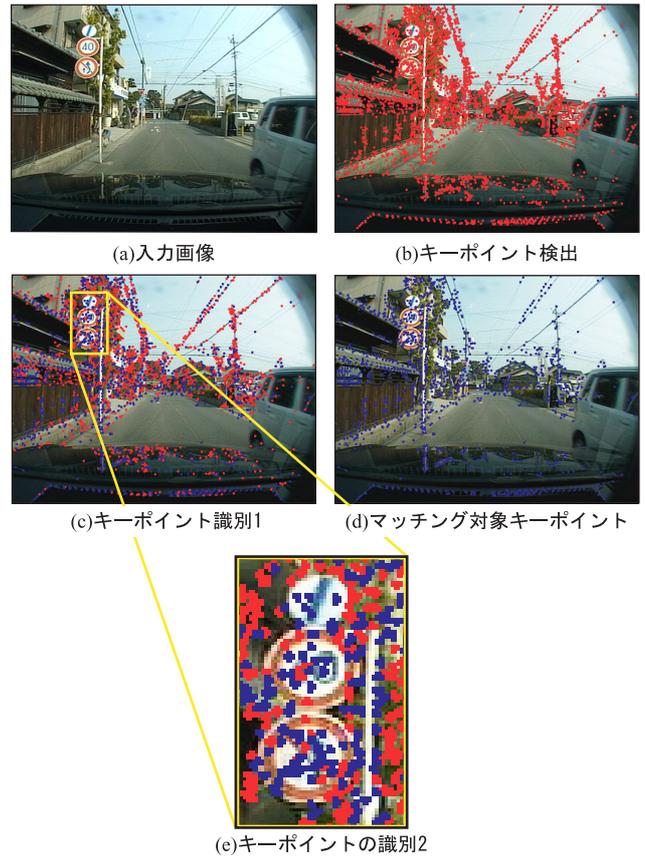


図 7 キーポイント識別例

れ用いて構築した Real AdaBoost 識別器による識別実験を行う。

4.1.1 実験概要

学習サンプルとしてテンプレート画像 (図 5 上部) を使用する。ポジティブクラス 41,919 点, ネガティブクラス 575,598 点のキーポイントを用いて学習を行う。評価用データには、部分空間の表現能力を比較するため学習サンプルと同じキーポイントを使用する。評価法として、横軸に誤識別率、縦軸に未識別率を両対数グラフに表した DET (Detection Error Tradeoff) カーブを使用する。グラフが原点に近ければ識別性能が高いことを表す。

4.1.2 キーポイント識別結果

図 8 に識別結果を示す。青線が SIFT, 緑線が PCA-SIFT, 赤線が提案手法である。橙色の点線を見ると、提案手法では誤識別率を 20% としたとき、PCA-SIFT に比べ、未識別率を約 10% 抑制することができた。これは、異なる部分空間を用いることで、PCA-SIFT では表現できなかった特徴空間を表現することができたからである。

図9に Real AdaBoost によって選択された特徴量次元数を示す。PCA を用いているため、高次元数に比べ低次元数の特徴量は特徴の分布が大きく、より特徴を表現している。従って、図9を見ると、学習開始時には特徴をより表現している寄与率が高い特徴量が選択されていることがわかる。提案手法では学習が進むと、一般画像クラスからだけでなく、標識画像クラスの特徴量を選択して学習が進んでいることがわかる。以上より、キーポイント識別には一般画像の部分空間と標識画像の部分空間にそれぞれ射影した特徴量を用いる提案手法が有効であることがわかる。

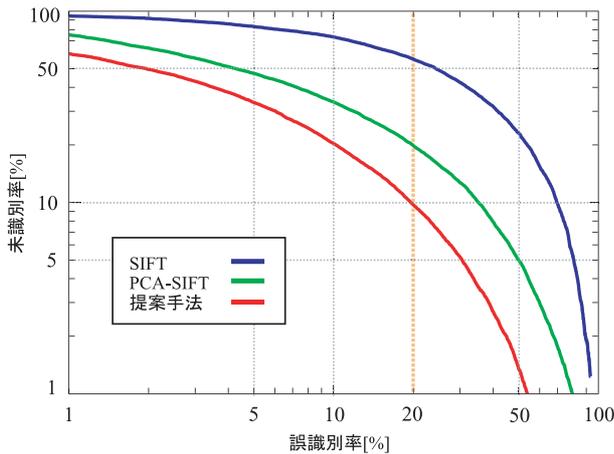


図8 識別結果

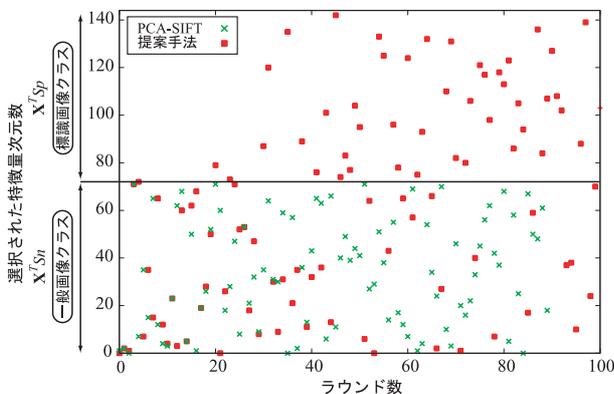


図9 選択された特徴量次元数

## 4.2 マッチング精度の比較

従来法と提案手法の対応点マッチングの精度とマッチング処理速度の比較実験を行う。

### 4.2.1 実験概要

車載カメラを用いて実環境下を撮影した映像全500フレームに対して処理を行い、従来法と提案手法の誤対応率と正解対応率を比較する。誤対応率を式(15)、正解対応率を式(16)により求める。

$$\text{誤対応率} = \frac{\text{誤対応数}}{\text{全マッチング数}} \quad (15)$$

$$\text{正解対応率} = \frac{\text{正解対応数}}{\text{全マッチング数}} \quad (16)$$

ここで誤対応数は、入力画像中の背景上のキーポイントと対応した数、正解対応数は標識上のキーポイントと正しく対応した数である。

### 4.2.2 マッチング精度結果

表1にマッチング精度の比較結果を示す。誤対応率を見ると、従来法に比べ約6.4%抑制していることが確認できる。さらに正解対応数では従来法に比べて少なくなっているが、全体としての割合を見ると、正解対応率が約6.4%向上した。これは、対象物体とそれ以外を表現できる特徴量を学習した識別器によって、背景上のキーポイントを削除しているためだと考えられる。図10にマッチング結果例を示す。特に下段に示す最高速度の標識において誤対応が抑制できているのがわかる。

表1 マッチング精度の比較

|         | 正解対応率 [%]         | 誤対応率 [%]            |
|---------|-------------------|---------------------|
| 従来法 [3] | 8.0<br>(163/2034) | 91.9<br>(1871/2034) |
| 提案手法    | 14.4<br>(39/269)  | 85.5<br>(230/269)   |

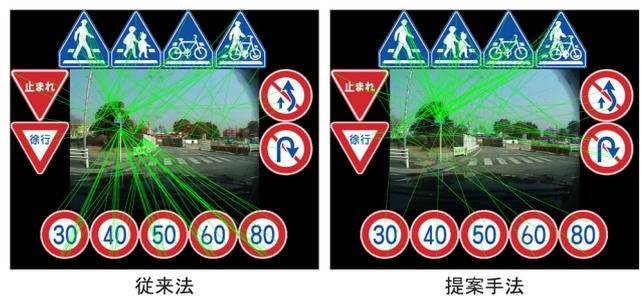


図10 マッチング例

### 4.3 マッチング処理時間の比較

従来法と提案手法のマッチング処理時間の比較を行う。

#### 4.3.1 実験概要

ある1フレームに対して従来法と提案手法のマッチング処理を計10回繰り返し、その平均時間を求め評価する。処理に用いるパーソナルコンピュータは、Intel

Xeon CPU 3.00GHz 8core を使用する．提案手法ではマッチング処理時間だけでなく，識別時間も含む．検出したキーポイント数は 2,150 点，識別器によって標識上のキーポイントと判断された点は 1,234 点である．

#### 4.4 マッチング処理時間結果

表 2 にマッチング処理時間の比較結果を示す．提案手法では識別処理を含んでいるのにも関わらず，マッチング処理時間を約 1/2 に短縮することができた．これはマッチング精度と同様に，予め，Real AdaBoost 識別器により，明らかに背景上のキーポイントを削除しマッチング対象のキーポイントを制限していること，それに加え，PCA を用いて特徴量次元数を減少させることでユークリッド距離の算出を高速に行うことができているからである．

表 2 処理時間 [ms]

|         | 識別時間 | マッチング時間 | 合計時間  |
|---------|------|---------|-------|
| 従来法 [3] | —    | 425.4   | 425.4 |
| 提案手法    | 20.3 | 182.5   | 202.8 |

## 5 標識認識

対応点マッチングの結果から標識認識を行う．標識認識には高木ら [3] によって提案された Voting 処理を用いる．

### 5.1 Voting 処理

標識認識には Voting 処理 (図 12) を用いる．まず，対応点マッチングで対応付けした対象標識画像 (図 13) 上のキーポイントから，その標識の基準点への位置ベクトルを求める．次に，入力画像上の対応付けしたキーポイントに対して，求めた位置ベクトルを用いて，式 (17) より，入力画像上の標識の基準点  $(X, Y)$  を投票する．ここで，テンプレート画像の特徴点の座標を  $(x_{temp}, y_{temp})$ ，スケールを  $\sigma_{temp}$ ，オリエンテーションを  $\theta_{temp}$ ，位置ベクトルを  $(\Delta x, \Delta y)$ ，入力画像の特徴点の座標を  $(x_{in}, y_{in})$ ，スケールを  $\sigma_{in}$ ，オリエンテーションを  $\theta_{in}$  とする． $\theta$  は  $\theta = \arctan(\Delta y / \Delta x)$  で求められる．投票した基準点が隣接している場合，それらの基準点をクラスタリングし投票数を求める．投票数がある閾値以上であれば標識が存在すると判定し認識を行う．

$$\begin{cases} X = x_{in} + \frac{\sigma_{in}}{\sigma_{temp}} \sqrt{\Delta x^2 + \Delta y^2} \times \cos(\theta + \theta_{temp} - \theta_{in}) \\ Y = y_{in} - \frac{\sigma_{in}}{\sigma_{temp}} \sqrt{\Delta x^2 + \Delta y^2} \times \sin(\theta + \theta_{temp} - \theta_{in}) \end{cases} \quad (17)$$

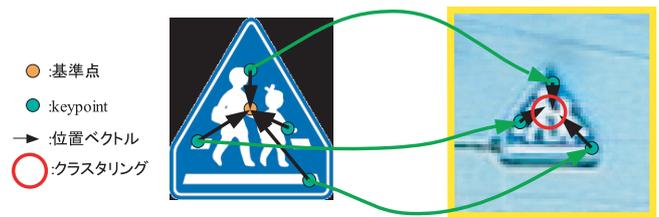


図 12 Voting 処理



図 13 認識対象標識

### 5.2 標識認識結果

図 11 に，標識認識例を示す．上が入力画像，中央が従来法，下が提案手法である．Real AdaBoost 識別器によって背景上のキーポイントに対応点マッチングの処理対象としていないため，従来法では認識できない標識を認識することができた．

## 6 おわりに

異なる部分空間における PCA-SIFT を用いた交通道路標識認識を提案し，異なる部分空間を用いることの有効性，標識認識における対応点マッチングの高速化を行った．異なる部分空間を用いることで，SIFT，PCA-SIFT，それぞれの特徴量を用いて構築した識別器と比較し，提案手法の識別器では誤識別率を 20% としたとき，未識別率を約 10% 抑制することができた．さらに，構築した識別器を用いて入力画像から検出したキーポイントの削除を行い，部分空間へ射影した特徴量を用いて対応点マッチングを行うことで対応点マッチングの精度の向上，マッチング処理速度の高速化を行った．

提案手法を用いて，車載カメラから撮影した標識を含む実画像での標識認識を行い，従来法では認識できなかった標識を認識することができることを確認した．今後は定量的評価を行うと共に，より識別性能の高い識別器の構築を行う予定である．



図 11 標識認識結果例

## 参考文献

- [1] 道満恵介, 高橋友和, 目加田慶人, 井手一郎, 村瀬洋, “カスケード型識別器による標識検出のための生成型学習法”, 画像の認識理解シンポジウム MIRU07, 講演予稿集, IS-5-04, pp.1396-14011, 2007
- [2] 景山陽一, 西田眞, 明珍甲太, “カラー情景画像における円形道路標識認識法”, 映像情報メディア学会誌, Vol.61, No.7, pp.972-978, 2007
- [3] 高木雅成, 藤吉弘亘, “SIFT 特徴量を用いた交通道路標識認”, 第 13 回画像センシングシンポジウム SSII07, LD2-06, 2007 .
- [4] D. Lowe, “Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision(IJCV), 60(2), pp. 91-110, (2004).
- [5] Y. Ke, R. Sukthankar, “PCA-SIFT: A more distinctive representation for local image descriptors”, Proc. of IEEE Conference on Computer Vision and Pattern Recognition, pp.511-517, 2004 .
- [6] J. J. Koenderink, “The structure of images”, Proc. of Biological Cybernetics, Vol.50, pp.363-373, 1984 .
- [7] T. Lindeberg, “Scale-space theory: A basic tool for analysing structures at different scales”, Proc of Journal of Applied Statistics, 21(2), pp.224-270 .
- [8] D. G. Lowe, “Object recognition from local scale-invariant keypoints”, Proc. of IEEE International Conference on Computer Vision, pp.1150-1157, 1999 .
- [9] [http://www.vision.caltech.edu/Image\\_Datasets/Caltech256/](http://www.vision.caltech.edu/Image_Datasets/Caltech256/)
- [10] R. E. Schapire and Y. Singer, “Improved Boosting Algorithms Using Confidence-rated Predictions”, Machine Learning, No.37, pp.297-336, 1999.