

2 段階の Randomized Trees を用いたキーポイントの分類

西村 孝[†] 清水 彰一[†] 藤吉 弘亘[†]

[†] 中部大学大学院工学研究科 〒 487-8501 愛知県春日井市松本町 1200
E-mail: [†] {takashi, shiyou}@vision.cs.chubu.ac.jp, hf@cs.chubu.ac.jp

あらまし 本稿では、2 段階の Randomized Trees による高精度かつ高速なキーポイントマッチング手法を提案する。従来法である Randomized Trees を用いたキーポイントの分類手法では、高精度なリアルタイムのキーポイントマッチングを実現している。しかし、Randomized Trees で表現するテンプレートの視点は多様であり、1 つの Randomized Trees では、全ての変化において高精度なキーポイントの分類は困難である。そこで、提案手法では 1 段階目にテンプレートの視点を分類し、2 段階目に、1 段階目で分類した視点ごとの Randomized Trees によりキーポイントの分類を行う。評価実験より、提案手法は対象物体の視点が 70 度回転した画像において、SIFT より 88.4%、Randomized Trees より 63.4% の精度を向上させることができた。また、提案手法は約 12 fps でリアルタイムにキーポイントマッチングが可能であることを確認した。

キーワード ランダム木、キーポイントマッチング、SIFT、視点分類

Keypoint Recognition using Two-Stage Randomized Trees

Takashi NISHIMURA[†], Shoichi SHIMIZU[†], and Hironobu FUJIYOSHI[†]

[†] Dept. of Computer Science, Chubu Univ. 1200, Matsumoto, Kasugai, Aichi, 487-8501 Japan
E-mail: [†] {takashi, shiyou}@vision.cs.chubu.ac.jp, hf@cs.chubu.ac.jp

Abstract This paper proposes a high-precision, high-speed keypoint matching method using a two-stage Randomized Trees. The keypoint classification method uses the conventional Randomized Trees to enable high-precision, real-time keypoint matching. But the wide variety of view transformations for templates expressed by Randomized Trees make high-precision keypoint classification for all transformations difficult with a single Randomized Trees. To resolve this problem, proposed method classifies the template view transformations during the first stage. Then during the second stage, it classifies the keypoints using the Randomized Trees corresponding to each of the view transformations classified during the first stage. For images in which the viewpoint of the object is rotated by 70°, evaluation testing demonstrated that proposed method is 88.4% more precise than SIFT, and 63.4% more precise than the conventional Randomized Trees. We have also shown that the proposed method supports real-time keypoint matching at a speed of 12 fps.

Key words Randomized Trees, Fast Keypoint Matching, SIFT, Viewpoint Classification

1. はじめに

画像から特定物体を自動的に認識する技術は、様々な分野において実現が望まれており、コンピュータビジョン分野において重要な研究テーマの一つである。ITS の分野では、ドライバの安全運転をサポートすることを目的とした交通道路標識の自動認識の研究が取り組まれている [1]。その他に、携帯電話のカメラで撮影した物体を自動的に認識し、商品を注文するシステムが提案されている [2]。このような特定物体の認識を実現するためには、画像の回転、スケール変化、照明変化、視点の変化などによる見えの変化に対して頑健な認識手法が必要であり、リアルタイムの処理も重視される。

近年では、見えの変化に対応するアプローチとして、画像の局所領域から得られる特徴量を用いたキーポイントマッチングが注目を集めている。局所特徴量を用いたキーポイントマッチングには、高性能な局所特徴量を用いた手法と、学習アルゴリズムを導入した 2 つの手法に分けることができる。前者の手法は、SIFT (Scale Invariant Feature Transform) [3] に代表されるような高性能な局所特徴量によるキーポイントマッチング手法である。SIFT は画像の回転、スケール変化、照明変化に頑健であるため、高精度なマッチングが可能である。近年では、より高精度なマッチングを実現するために、SIFT の記述能力を高性能化した PCA-SIFT [4] や GLOH [5]、Shape Context [6]、ASIFT [7] が提案されている。しか

し、このような SIFT ベースの手法は計算コストが高いという問題がある．SIFT を高速化した SURF [8] や Fast approximated SIFT [9] も提案されているが、現状ではリアルタイム処理が困難である．

一方、学習アルゴリズムを導入した手法では、各キーポイントをクラスとし、クラスに分類するという問題に置き換えることにより、キーポイントマッチングを実現している．その手法の一つとして、オフライン処理によりキーポイントを分類する Randomized Trees を学習し、オンライン処理時に、学習した Randomized Trees を用いてキーポイントを分類する手法が提案されている [10]．文献 [10] は、1 枚のテンプレート画像から、アフィン変換によってテンプレートの多様な見えの変化を擬似的に表現した学習画像を生成し、Randomized Trees [11] により学習する．これにより、見えの変化に対して頑健なキーポイントの分類が可能となる．また、Randomized Trees は対応点探索問題を決定木のトラバースルにより実現するため、高速にキーポイントを分類することができる．そのため、リアルタイムに特定物体を認識するタスクに有効的な手法である．近年では、この手法をさらに発展させ、実装メモリが小さいモバイルデバイスでも動作可能にしたという報告もある [12], [13]．しかし、文献 [11] をベースとした手法は、画像の見えの変化が大きい場合にマッチング精度が低下する問題がある．この原因として、一つの Randomized Trees で表現するテンプレートの見えの変化が多様であることが考えられ、全ての見えの変化において高精度なキーポイントの分類が困難となる．

そこで、この問題を解決するために、本研究では 2 段階の Randomized Trees を用いることにより、見えの変化に頑健なキーポイント分類法を提案する．提案手法では、1 段階目に入力画像の視点の分類を行い、2 段階目では、1 段階目で分類した近い視点の画像で学習した Randomized Trees によりキーポイントの分類を行う．これにより、入力画像の見えに近い画像で学習した Randomized Trees を用いたキーポイントの分類が可能となるため、キーポイントの分類性能の向上が期待できる．

2. Randomized Trees によるキーポイントの分類

本章では、本研究の基礎となる文献 [10] について説明する．文献 [10] は、学習アルゴリズムを導入したキーポイント分類法である．学習には、Randomized Trees [11] を利用しており、高速にキーポイントの分類が可能である．また、学習データにアフィン変換した画像を用いることで、見えの変化に対する頑健を得ている．

2.1 キーポイントの検出

キーポイントマッチングに必要なキーポイントの検

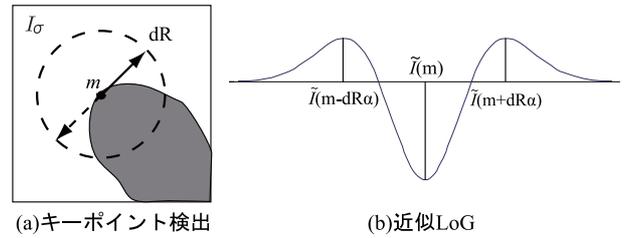


図 1 近似 LoG によるキーポイント検出

出 [14] を行う．キーポイントの検出は、図 1(a) に示すように、テンプレートをラスタ走査し注目画素 $m = (x, y)$ の LoG の応答値を算出する．そして、注目画素の 8 近傍で最大応答値となる注目画素 m をキーポイントの候補とする．文献 [10] では、LoG を高速に計算するために式 (1) で近似する．近似 LoG のイメージを図 1(b) に示す．

$$\text{LoG}(\mathbf{m}) \approx \sum_{\alpha \in [0; \pi]} I_{\sigma}(\mathbf{m} - d\mathbf{R}_{\alpha}) - I_{\sigma}(\mathbf{m}) + I_{\sigma}(\mathbf{m} + d\mathbf{R}_{\alpha}) \quad (1)$$

ここで、 I_{σ} は平滑化後の画像である． $d\mathbf{R}_{\alpha} = (R \cos \alpha; R \sin \alpha)$ であり、 R は半径を示す． R の大きさを変化させ、応答値が最大になったときの R をキーポイントのスケール値とする．

次に、画像の見えの変化に対する頑健性を得るために、算出したキーポイント候補からアフィン変化に頑健なキーポイントを取捨選択する．テンプレートを複数のアフィンパラメータでアフィン変換し、全てのアフィン変換画像でキーポイント候補を検出する．複数のアフィン変換画像で多く検出することができた同一のキーポイント候補のみを選択することで、アフィン変化に頑健なキーポイントを算出することができる．次に、算出されたキーポイントを用いて決定木を学習する．

2.2 Randomized Trees

Randomized Trees は、マルチクラスの識別問題に対応した学習手法の一つである．Randomized Trees は、学習サンプルに含まれるノイズに対して頑健な特長があり、高い識別精度でありながら高速に処理できるため、近年注目を集めている．文献 [10] では、Randomized Trees をキーポイントの分類問題に適用している．以下に、Randomized Trees の構築方法について述べる．

2.2.1 学習

各決定木を学習するために、 L 個のサブセットを作成する．サブセットはランダムに選択されたキーポイントを中心とした 32×32 ピクセルのパッチからなる．決定木は 1 つのサブセットから学習する．

・決定木

決定木 $T = \{T_1, \dots, T_L\}$ は分岐点 (ノード) とノードの末端 (リーフノード) から構成される．ノードには子ノードに分岐するための条件 (スプリット関数) が用意

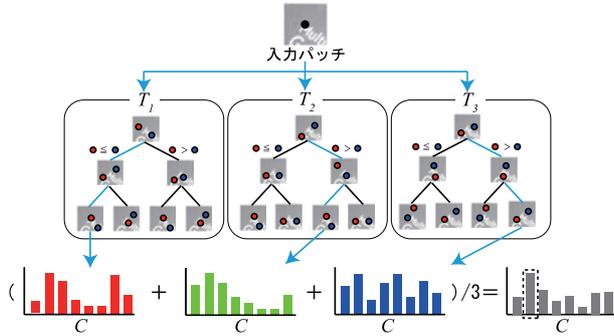


図 2 キーポイントの分類 [10]

されている．スプリット関数とノードの階層数は事前に決定する．

・ノード

ノードのスプリット関数は，パッチからランダムに選択した 2 点のピクセル m_1 と m_2 の輝度の大小関係の特徴量として，式 (2) に示すように分岐先を決定する．

$$C_2(m_1, m_2) = \begin{cases} L & \text{If } I_\sigma(\mathbf{P}, m_1) \leq I_\sigma(\mathbf{P}, m_2) \\ R & \text{otherwise} \end{cases} \quad (2)$$

L は左側，R は右側の子ノードに分岐することを示している． $I_\sigma(\mathbf{P}, m)$ は，パッチ \mathbf{P} における注目画素 m の輝度値である．この特徴は 2 点の輝度差を捉えており，2 点の輝度の大小関係が変わらない限り，スプリット関数の出力が変化しないため，照明変化に頑健となる．ノードの分岐は，事前に決めたノードの階層数に到達すると終了する．末端のノードをリーフノードと呼び，各キーポイントの確率分布を持つ．

・リーフノード

リーフノードは，到達した各キーポイントの確率分布 $P_{\eta(T_i, \mathbf{P})}(Y(\mathbf{P}) = c)$ を持つ． $\eta(T_i, \mathbf{P})$ は，入力パッチ \mathbf{P} の到達した決定木 T_i のリーフノードであり， c はテンプレートのキーポイントの種類を示す．

2.2.2 分類

全てのサブセットから学習した決定木群 $T = \{T_1, \dots, T_L\}$ によりキーポイントを分類する．分類の流れを図 2 に示す．入力画像からキーポイントを検出し，各キーポイントから 32×32 ピクセルのパッチを作成する．キーポイントのパッチを決定木に入力し，トラバースルすることで各キーポイントのリーフノードの確率分布 $P_{\eta(T_i, \mathbf{P})}(Y(\mathbf{P}) = c)$ を得る．そして，式 (3) のように平均確率が最大となるキーポイントをキーポイント分類結果として出力する．

$$\hat{Y}(\mathbf{P}) = \arg \max_c \frac{1}{L} \sum_{l=1}^L P_{\eta(T_l, \mathbf{P})}(Y(\mathbf{P}) = c) \quad (3)$$

2.3 問題点

Randomized Trees を用いたキーポイントの分類手法は，全てのアフィン変化を含んだ学習画像で Randomized

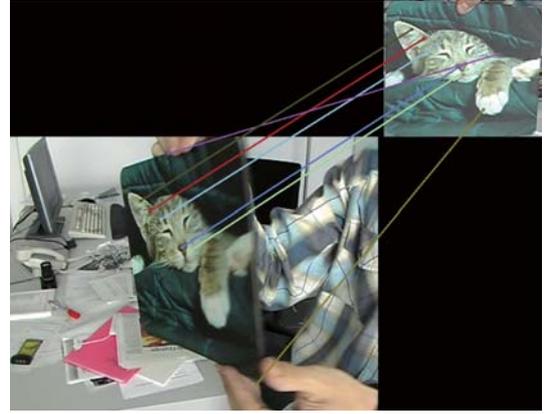


図 3 文献 [10] によるキーポイントマッチング結果

Trees を学習する．そのため，同一のキーポイントであっても多様な見えを含んでおり，1 つの Randomized Trees で正確に表現するのは困難である．また，学習画像の生成にアフィン変換を用いているため，3 次元上の回転が表現できないことや，アフィン変換のパラメータをランダムで決定しているため，生成される学習画像の視点が偏るという問題が生じる．そのため，図 3 に示すようにマッチング精度が低下する場合がある．この問題を解決するために，入力画像の見えの変化とキーポイントの分類問題を 2 段階の Randomized Trees で表現する．さらに，学習画像の生成では，オイラー角を用いて 3 次元上の回転を表現する．そして，XYZ 軸の各回転パラメータを等間隔で設定することにより，視点の偏り問題を解決する．以下に，提案手法について述べる．

3. 提案手法: 2 段階の Randomized Trees によるキーポイントマッチング

提案手法は，テンプレートの視点の変化とキーポイントの分類問題に対応するために，2 段階の Randomized Trees を学習する．1 段階目では入力画像の Viewpoint クラスを分類する．Viewpoint クラスとは，学習画像の多様な視点を K 個の代表的な視点にクラスタリングしたものである．2 段階目は，1 段階目で分類した Viewpoint クラスに属する学習画像により学習した Randomized Trees によりキーポイントの分類を行う．これにより，入力画像に近い視点の画像で学習した Randomized Trees によるキーポイントの分類ができる．提案手法の処理の流れを図 4 に示す．はじめに，前処理としてテンプレートの多様な視点を表現した学習画像を生成し，視点を Viewpoint クラスタリングする．次に，2 段階の Randomized Trees を学習する．

3.1 学習画像の生成

学習画像の生成と Viewpoint クラスタリングを行う．以下にその詳細について述べる．

3.1.1 3 次元回転学習画像

文献 [10] では，学習画像を生成する際に，アフィン変

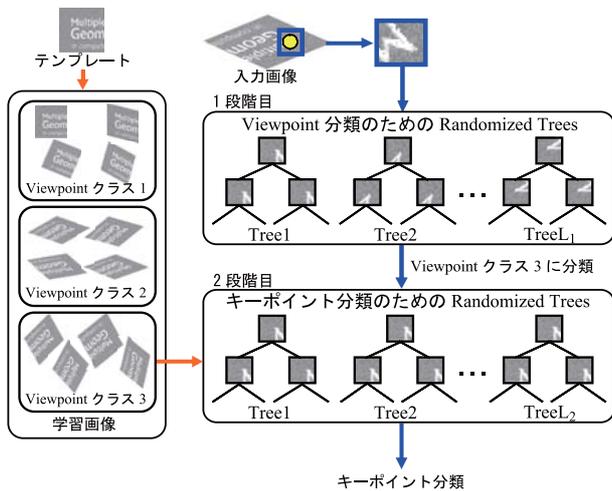


図 4 提案手法の処理の流れ

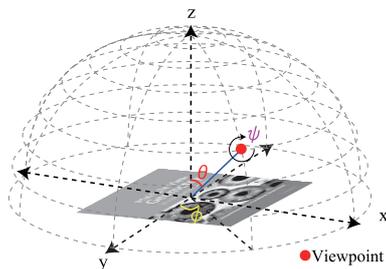


図 5 Viewpoint

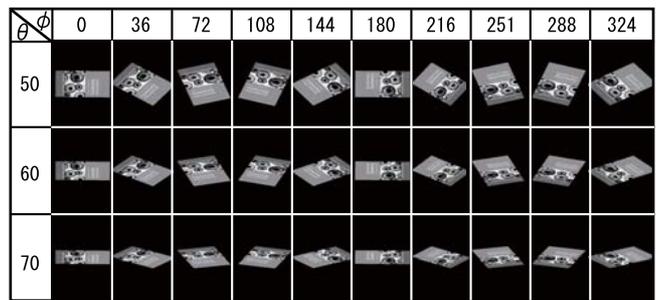
換のパラメータをランダムに選択している．そのため、視点の偏りが発生する問題や 3 次元上の回転を表現できないという問題がある．この問題を解決するために、提案手法では学習画像の生成にオイラー角を用いる．これにより、図 5 に示すように、3 次元上の回転を表現することができる．2 次元座標上におけるオイラー角を用いたアフィン変換行列 A を式 (4) とする．

$$A = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix}$$

ここで、 A は 2 次元座標に変換する 2×2 の行列である．学習画像を生成する際には、オイラー角の ψ, θ, ϕ の回転パラメータを等間隔に設定したアフィン変換行列 A から求め、視点の偏り問題を解決する．本研究では、各パラメータの回転範囲を $\phi \in [0^\circ, 90^\circ], \theta \in [0^\circ, 90^\circ], \psi \in [0^\circ, 360^\circ]$ とし、 ϕ, θ, ψ の間隔は 5° とする．このとき、1 枚のテンプレート画像から学習画像 23328 枚を生成することになる．図 6 に $\phi = 90^\circ$ のときの θ と ψ を変化させて生成した画像例を示す．

3.1.2 Viewpoint クラスタリング

次に Viewpoint クラスタリングを行う．オイラー角の XYZ 軸の回転パラメータによりクラスタリングすると、回転における周期性を表現することができない．そこで、提案手法では生成したパッチ画像を特徴量として k-means 法により視点をクラスタリングする．これにより、画像の回転が 0° と 359° のときでも、視点の近い画



$\psi=90$

図 6 学習画像の生成例

像としてクラスタリングできる．クラスタリングの流れを図 7 に示す．学習画像ごとにキーポイントを中心とする 32×32 ピクセルのパッチを結合したパッチ画像列を作成する．そして、パッチ画像列を輝度の特徴空間に投影して k-means によりクラスタリングする．Viewpoint クラス数 K を 30 としたときのクラスタリング結果を各クラスで色分けした例を図 8 に示す．図 8 より、視点の近いものをクラスタリングできていることがわかる．図 9 に、各クラスに含まれる学習画像を示す．図 9 から、視点が近い画像を同一クラスにクラスタリングできていることがわかる．

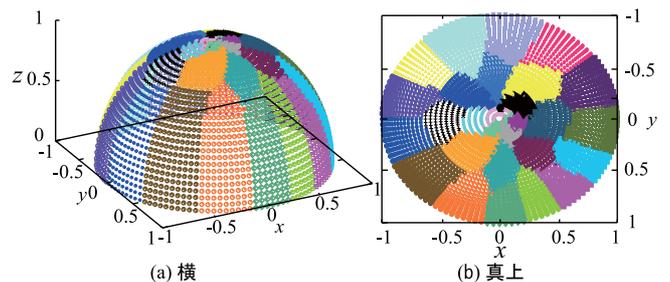


図 8 Viewpoint クラスタリング結果の球面表示

3.2 2 段階の Randomized Trees の学習

提案手法は、テンプレートの視点の変化とキーポイントの分類問題に対応するために、2 段階の Randomized Trees を学習する．1 段階目の Randomized Trees では、パッチの輝度値の大小関係を用いて、Viewpoint クラスの頻度を Randomized Trees で学習する．2 段階目の Randomized Trees では、Viewpoint クラスごとに Randomized Trees を学習する．したがって、図 8 の色ごとに示した Viewpoint クラスごとに Randomized Trees を作成する．2 段階の Randomized Trees の学習方法の詳細を以下に述べる．

3.2.1 1 段階目：Viewpoint クラス分類の Randomized Trees の学習

入力画像の Viewpoint クラスを分類する決定木群 $T1 = \{T1_1, \dots, T1_{L_1}\}$ を学習する． L_1 は決定木の数を示す．決定木群 $T1$ は、パッチを L_1 個のサブセットに分けて学習する、ノードの分岐は、式 (2) と同様にキー

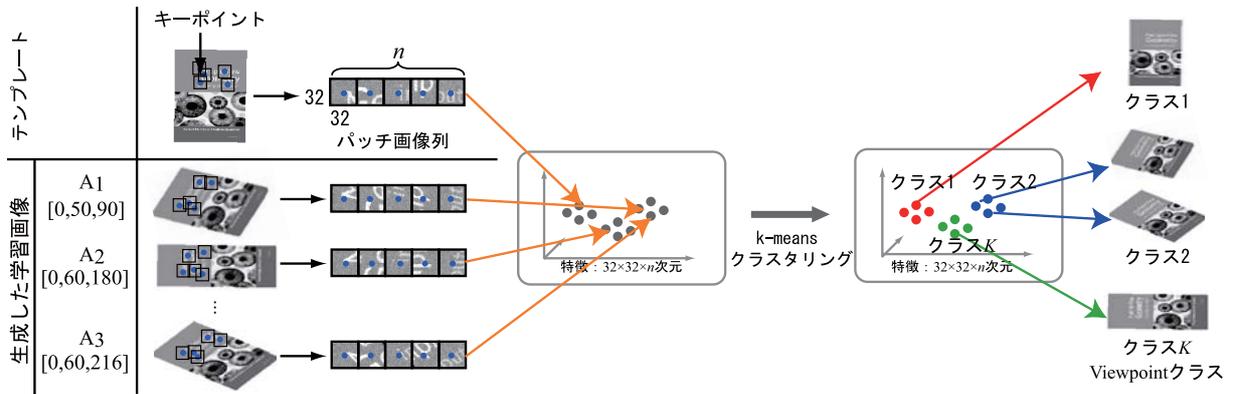


図 7 Viewpoint のクラスタリング

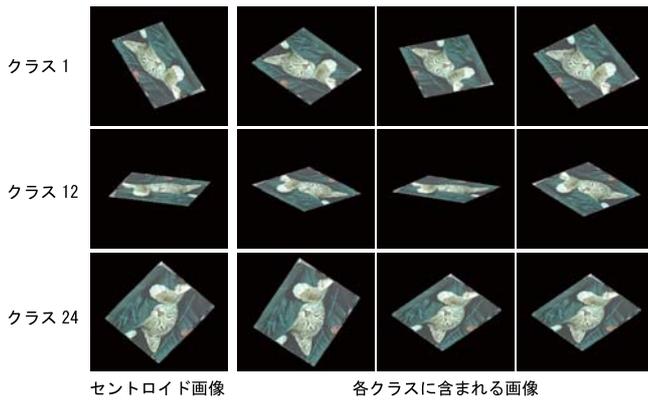


図 9 クラスタリング結果

ポイントのパッチの輝度値の大小関係により決定する。そして、リーフノードの Viewpoint クラスの確率分布を求める。これにより、分類時にはパッチが到達したリーフノードの確率分布から Viewpoint クラスを分類することができる。本研究では 1 段階目の Randomized Trees で取り扱うパッチ数は、学習画像数 $m = 23328$ ，キーポイントクラス数 $c = 400$ とした場合では 933 万個となる。

3.2.2 2段階目:キーポイント分類の Randomized Trees の学習

Viewpoint クラスごとにキーポイントを分類する決定木群を学習する。2 段階目の決定木群は K 個の Viewpoint クラスの決定木群の集まり $T2 = \{T2_1, \dots, T2_K\}$ から構成される。Viewpoint クラス $k (k \in K)$ の決定木群 $T2_k = \{T2_{k,1}, \dots, T2_{k,L_2}\}$ は、Viewpoint クラス k に属するパッチを L_2 個のサブセットに分けて学習する。ノードの分岐は、式 (2) と同様にキーポイントのパッチの輝度値の大小関係により決定する。そして、リーフノードのキーポイントクラスの確率分布を求める。本研究では 2 段階目の $T2_k$ の Randomized Trees で取り扱うパッチ数は、学習画像数 $m = 23328$ ，キーポイントクラス数 $c = 400$ ，Viewpoint クラス数 $K = 30$ とした場合では 31 万個となる。

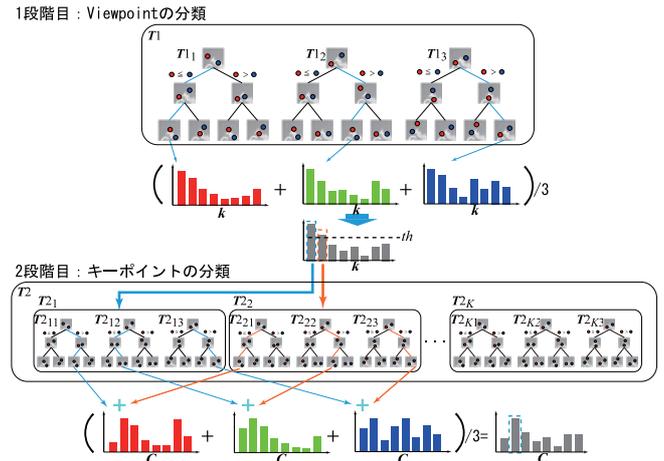


図 10 2 段階の決定木によるキーポイントの分類

3.3 2 段階の Randomized Trees を用いたキーポイントの分類

キーポイントの分類の流れを図 10 に示す。1 段階目の決定木群 $T1$ で入力画像の Viewpoint クラス k を分類する。次に、2 段階目の決定木群の集まりである $T2$ から、1 段階目で分類した Viewpoint クラス k に対応する決定木群 $T2_k$ によりキーポイントを分類する。

3.3.1 1 段階目: Viewpoint クラスの分類

Viewpoint クラス k の分類は、テンプレート上のキーポイント全体を用いて式 (4) に示すように、入力パッチ P が到達した決定木 $T1 = \{T1_1, \dots, T1_{L_1}\}$ のリーフノ

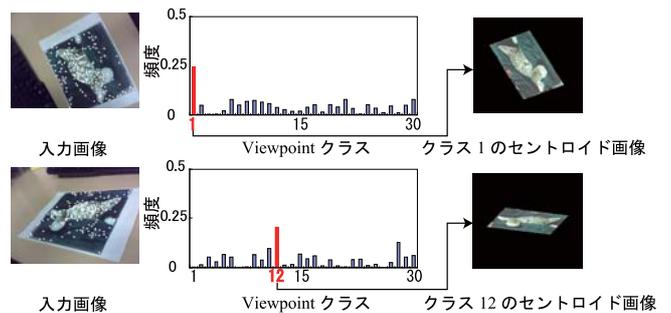


図 11 Viewpoint クラスの分類例

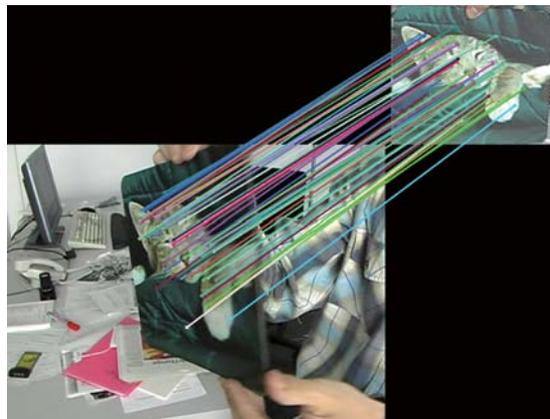
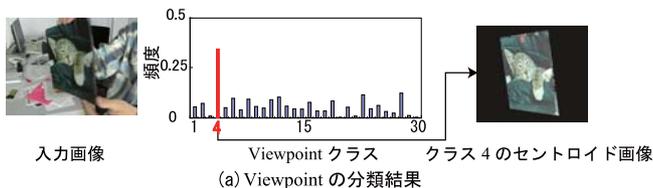


図 12 キーポイントマッチング例

ドの確率分布 $P_{\eta(T_{l_1, \mathbf{P}})}(Y(\mathbf{P}) = k)$ の平均を求め、しきい値 th を超える Viewpoint クラス k を抽出する。

$$G(k) = \begin{cases} 1 & \text{If } \frac{1}{L_1} \sum_{l=1}^{L_1} P_{\eta(T_{l_1, \mathbf{P}})}(Y(\mathbf{P}) = k) > th \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

図 11 に Viewpoint クラスの分類例を示す。入力画像と最も頻度の高い Viewpoint クラスのセントロイド画像が似ていることがわかる。

3.3.2 2段階目：キーポイントの分類

1段階目の Viewpoint クラスの分類では、 $G(k) = 1$ となるクラスが複数存在する場合がある。そのためキーポイントの分類は、 $G(k) = 1$ となる複数の決定木群 T_{2_k} から、リーフノードの確率分布 $P_{\eta(T_{2_{kl}, \mathbf{P}})}(Y(\mathbf{P}) = c)$ の平均を求め、式 (5) より確率の高いキーポイントのクラス c に分類する。

$$\hat{Y}(\mathbf{P}) = \arg \max_c \frac{1}{L_2} \sum_{l=1}^{L_2} \sum_{k=1}^K G(k) P_{\eta(T_{2_{kl}, \mathbf{P}})}(Y(\mathbf{P}) = c) \quad (5)$$

提案手法は、図 12(a) のように、1段階目の Randomized Trees で入力画像の Viewpoint クラスの分類を行い、2段階目では、クラス 4 のセントロイド画像の視点に近い学習画像を用いて学習した Randomized Trees を用いるため、高精度なキーポイントマッチングが可能となる。

4. 評価実験

提案手法のマッチング精度は、Viewpoint クラス数と大きな関係がある。そこで、はじめに、Viewpoint クラス数によるマッチング精度の関係を評価する。さらに、

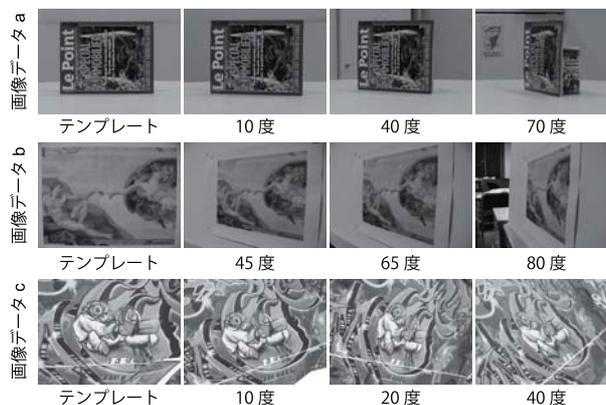


図 13 データベース

Viewpoint クラスは手動でも与えることができるため、手動によるクラスタリングと提案手法によるクラスタリングを用いたときのマッチング精度の比較を行う。最後に、提案手法の有効性を示すために、テンプレートのキーポイントマッチング実験により従来法との精度の比較を行う。また、処理時間の比較も行う。

4.1 データベース

実験には Mikolajczyk database [15] と Morel database [16] を用いる。各データベースの画像データを図 13 に示す。Morel database の画像データ a は、10度～70度まで物体を回転して撮影した画像 7 枚、Morel database の画像データ b は、45度～80度まで物体を回転した画像 4 枚、Mikolajczyk database の画像データ c では、10度～40度まで物体を回転した画像 3 枚である。

4.2 実験概要

比較対象は、SIFT [3]、Randomized Trees (RTs) [10]、ASIFT [7] である。式 (6) から得られるマッチング精度により比較する。

$$\text{マッチング精度} = \frac{\text{マッチング成功数}}{\text{マッチング数}} \quad (6)$$

SIFT では、オクターブが 6、DOG 画像を 4 つとする。Randomized Trees の学習では、学習画像を 23328 枚、1段階目と 2段階目の Randomized Trees のサブセット数 L_1, L_2 を 30 個、決定木の深さを 15 階層とする。また ASIFT は、プログラム [16] をダウンロードし、添付のバッチファイルを使用する。

4.3 Viewpoint クラスタリングの精度評価

提案手法は、Viewpoint クラス数により精度が変わるため、Viewpoint クラス数を 5～40 を 5 刻みで変化させたときのマッチング精度を比較する。実験結果を図 14 に示す。Viewpoint クラス数を増やすと精度は向上しているが、30 個を境にマッチング精度が飽和しているのがわかる。結果より Viewpoint クラス数は 30 個程度が最適であるといえる。

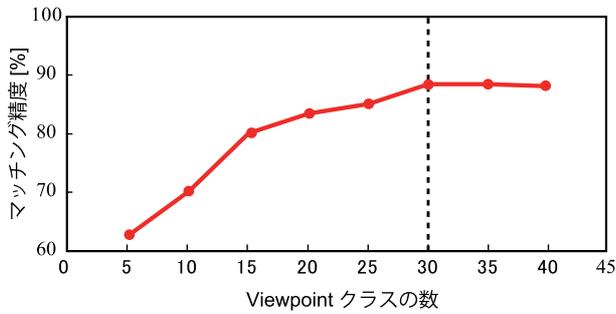


図 14 Viewpoint クラスの数による精度比較結果

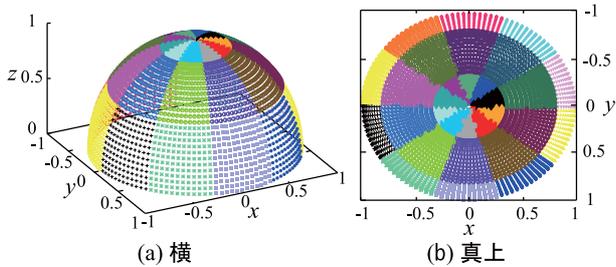


図 15 手動による Viewpoint クラスタリング結果

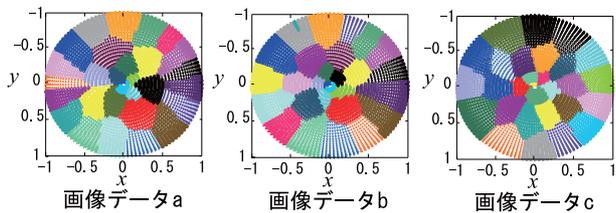


図 16 各テンプレートの Viewpoint クラスタリング結果

表 1 クラスタリングによるキーポイントマッチング精度 [%]

	手動	自動
画像データ a	86.9	88.5
画像データ b	82.6	84.6
画像データ c	85.1	86.4
平均	84.8	86.5

次に、手動によるクラスタリングと提案手法によるクラスタリングを用いたときのマッチング精度を比較する。手動によるクラスタリングは、図 15 に示すように等間隔に 30 個のクラスに分割する。図 16 に各テンプレートの提案手法による Viewpoint クラスタリング結果を示す。表 1 にキーポイントマッチング精度を示す。表 1 より、全てのテンプレートにおいて提案手法のクラスタリングを用いると精度が向上しているのがわかる。これは、輝度値でクラスタリングすることで、テンプレートのテクスチャに依存する輝度値の変化が大きい視点や変化の少ない視点を考慮した Viewpoint クラスを生成できているためである。

4.4 精度と処理速度

提案手法の有効性を評価するためにマッチング精度と処理速度の比較実験を行う。処理速度の測定に用いた PC スペックは XeonX5355@2.66 GHz である。

各画像データにおけるマッチング精度を表 2～表 4 に

表 2 画像データ a のマッチング精度 [%]

	10	20	30	40	50	60	70	平均
提案手法	100	100	100	100	99.4	100	88.5	98.2
SIFT	100	100	100	99.3	96.9	77.8	0	82.0
RTs	100	100	100	100	98.5	98.0	25.0	88.7
ASIFT	100	100	100	100	100	100	100	100

表 3 画像データ b のマッチング精度 [%]

	45	65	75	80	平均
提案手法	100	100	100	84.6	97.4
SIFT	82.8	77.1	48.5	0	52.1
RTs	100	100	100	25.0	82.5
ASIFT	100	100	100	100	100

表 4 画像データ c のマッチング精度 [%]

	10	20	40	平均
提案手法	100	93.1	86.4	93.1
SIFT	100	88.6	37.9	75.5
RTs	100	91.8	50.0	80.6
ASIFT	100	100	100	100

表 5 処理時間の計測

	処理時間 [ms]
提案手法	78
SIFT	1221
RTs	45
ASIFT	48212

示し、図 17 に各画像データのキーポイントマッチング結果画像を示す。また、処理速度の測定結果を表 5 に示す。

表 2～表 4 から、マッチング精度の最も精度が高い手法は ASIFT であり、次に提案手法、Randomized Trees、SIFT の順である。また、処理速度は、Randomized Trees、提案手法、SIFT、ASIFT である。マッチング精度と処理速度の両方を考慮して提案手法と ASIFT を比較すると、提案手法は ASIFT に比べ平均 3.8 %精度が低下しているが、提案手法の処理時間は ASIFT の 0.16 %程であり、提案手法は高速なキーポイントマッチングを実現している。従来手法である Randomized Trees と提案手法を比べると、提案手法は処理速度が 23 ms 増加しているが、12 fps でキーポイントマッチングが可能である。また、マッチング精度では平均 12.3 %精度を向上させることができた。これは、2 段階に Randomized Trees を学習することで、テンプレートの多様な視点を 1 段階目で限定することが可能となり、2 段階目の Randomized Trees では、キーポイントの分類問題を簡易化することができたため精度が向上したと考えられる。

図 18 にマッチング精度と処理時間の関係を示す。図 18 は、左上に分布するほど、高精度かつ高速な手法であることを示す。提案手法は最も左上に分布していることから、アフィン変化を伴うシーンにおいてリアルタイム処理を必要とする場合に有効であるといえる。

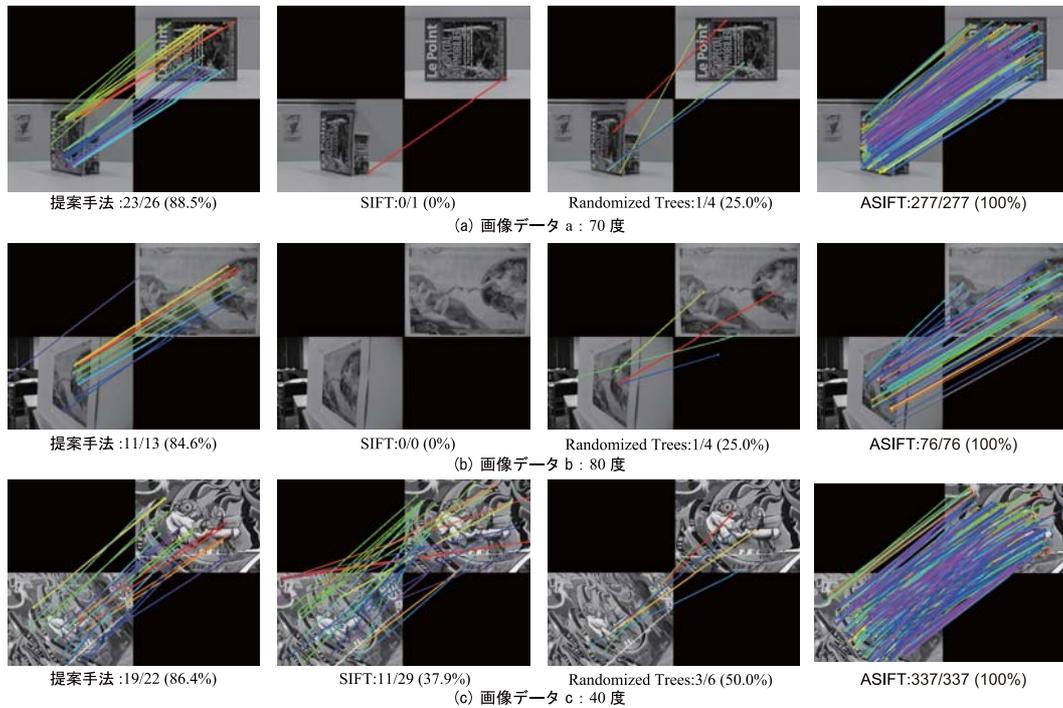


図 17 各画像データのキーポイントマッチング結果

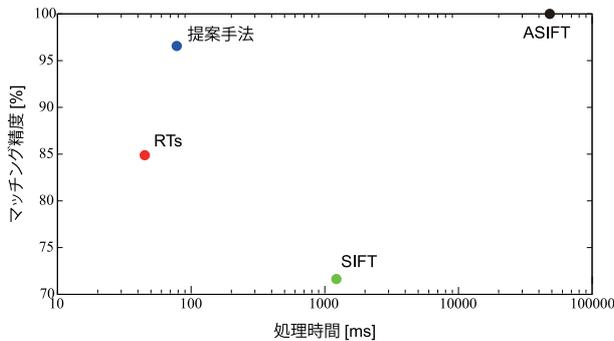


図 18 各手法の処理速度と精度の関係

5. おわりに

本稿では、2段階の Randomized Trees を用いたキーポイントの分類手法を提案した。提案手法は、テンプレートの視点の変化とキーポイントの分類の2つの問題を2段階の Randomized Trees で解決した。そのため、従来法の Randomized Trees に比べて分類問題を簡易化することが可能となった。その結果、対象物体の視点70度回転した入力画像において、SIFTより88.4%、Randomized Treesより63.4%の精度向上をさせることができた。さらに、提案手法では12fpsでキーポイントマッチングが可能であることを確認した。今後は、メモリ量を削減した Randomized Trees の学習や、オンラインで学習する手法を検討する予定である。

文 献

- [1] 高木雅成, 藤吉弘巨: “SIFT 特徴量を用いた交通道路標識認識”, 電気学会論文誌, **129-C**, 5, pp. 824–831 (2009).
- [2] 野口和人, 氏原慎弥, 黄瀬浩一, 岩村雅一: “カメラ付き携帯電話を入力デバイスとした大規模画像認識”, 情報処理学会研究報告 CVIM, **29**, pp. 205–210 (2009).
- [3] D. G. Lowe: “Distinctive image features from scale-invariant keypoints”, *Int. Journal of Computer Vision*, **60**, pp. 91–110 (2004).
- [4] Y. Ke and R. Sukthankar: “PCA-SIFT: A more distinctive representation for local image descriptors”, *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, **2**, pp. 506–513 (2004).
- [5] K. Mikolajczyk and C. Schmid: “A performance evaluation of local descriptors”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 10, pp. 35–47 (2005).
- [6] S. Belongie, J. Malik and J. Puzicha: “Shape matching and object recognition using shape contexts”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, 4, pp. 509–522 (2002).
- [7] J. M. Morel and G. Yu: “Asift: A new framework for fully affine invariant image comparison”, *SIAM Journal on Imaging Sciences*, **2**, 2, pp. 438–469 (2009).
- [8] H. Bay, T. Tuytelaars and L. V. Gool: “SURF: speeded-up robust features”, In *ECCV*, pp. 404–417 (2006).
- [9] G. Michael, G. Helmut and B. Horst: “Fast approximated SIFT”, *Proc. of ACCV*, pp. 918–927 (2006).
- [10] V. Lepetit and P. Fua: “Keypoint recognition using randomized trees”, *Transactions on Pattern Analysis and Machine Intelligence*, **28**, 9, pp. 1465–1479 (2006).
- [11] L. Breiman: “Random forests”, *Machine Learning*, **45**(1), pp. 5–32 (2001).
- [12] M. Ozuyosal, M. Calonder, V. Lepetit and P. Fua: “Fast keypoint recognition using random ferns”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2009).
- [13] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond and D. Schmalstieg: “Pose tracking from natural features on mobile phones”, *Proc. ISMAR 2008* (2008).
- [14] E. Rosten and T. Drummond: “Machine learning for high-speed corner detection”, In *European Conference on Computer Vision*, pp. 430–443 (2006).
- [15] <http://www.robots.ox.ac.uk/~vgg/research/affine/>.
- [16] <http://www.cmap.polytechnique.fr/~yu/research/ASIFT/>.