

Cascaded FAST によるキーポイント検出

長谷川 昂宏^{†a)} 山内 悠嗣^{†b)} 安倍 満^{††c)} 吉田 悠一^{††d)}
 山下 隆義^{†e)} 藤吉 弘亘^{†f)}

Keypoint Detection by Cascaded FAST

Takahiro HASEGAWA^{†a)}, Yuji YAMAUCHI^{†b)}, Mitsuru AMBAI^{††c)},
 Yuichi YOSHIDA^{††d)}, Takayoshi YAMASHITA^{†e)}, and Hironobu FUJIYOSHI^{†f)}

あらまし 本論文では、自然領域からのコーナー点の検出を抑制することで高速なコーナー検出及び 2 画像間の対応づけが可能な Cascaded FAST を提案する。複雑なテクスチャ（木の葉、植え込みなどが写り込んでいる領域）を含む画像に、高速なコーナー検出が可能な FAST を適用すると多くのコーナーが検出されるが、これらは物体認識等のアプリケーションでは必要としない。これは、FAST では各注目画素においてわずか周囲長 16 画素の同心円の輝度情報に基づいてコーナーを定義しているためである。そこで、提案手法では各注目画素において周囲長が 20 画素、16 画素、12 画素の 3 種類の同心円を配置させ、各同心円の輝度値の連続性とオリエンテーションの類似性によりコーナーを定義する。また、FAST のアプローチで学習した三つの決定木をカスケード状に並べることで、非コーナーを早期棄却する高速なコーナー検出を実現する。更に、提案手法では画像ピラミッドを用いることでスケールを獲得し、周囲の画素を参照する枠組みを利用することで高速にオリエンテーションを獲得する。実験により、Cascaded FAST は高速にキーポイントを検出しながらも自然領域からのキーポイント検出を大幅に抑制できたことを確認した。2 画像間の対応付けの実験では従来のコーナー検出法と同等の性能を保ちながら高速に 2 画像間に対応付けることを確認した。

キーワード コーナー検出, Cascaded FAST, FAST, キーポイントマッチング

1. ま え が き

画像間の特徴点（キーポイント）の対応付け問題は、特定物体認識、三次元地図作成、パノラマ画像生成、画像検索などのアプリケーションにおいて重要な課題である。画像間の対応付けは、画像中からキーポイントを検出した後、キーポイントに対して特徴量を記述し、各キーポイントの特徴量を比較することで実現できる。キーポイント検出は画像間の対応付けの前処理

に相当するタスクであるため、可能な限りキーポイント検出が高速であることが望ましい。

画像間のキーポイントを対応付けるためには、見えの変化が生じた複数の画像において同じ点が同じ位置で安定して検出される必要がある。この理由から、現在までに多くのキーポイント検出法が提案され、これらの手法はコーナー検出法 (Harris [1], FAST [2]), ブローブ検出法 (DoG [3], Harris-Affine [4]), 領域検出法 (MSER [5]) の三つのアプローチに分類される [6]。本研究は、高速なキーポイント検出を目的とするため、コーナー検出法に分類される手法を提案する。コーナー検出は 1981 年に提案された Moravec のコーナー検出法 [7] 以降、その研究の重要性から多くの手法 [1], [8], [9] が提案された。近年では、高速なコーナー検出が可能な Features from Accelerated Segment Test (FAST) [2] が注目されている。FAST は注目画素と注目画素を中心とする周囲長 16 画素の輝度値の大小関係を用いてコーナーを判定する。そして、この定義に基づいて機械学習により学習した決定木を

[†] 中部大学大学院工学研究科, 春日井市
 Graduate School of Engineering, Chubu University, 1200
 Matsumoto-cho, Kasugai-shi, 487-8501 Japan

^{††} (株) デンソーアイティラボラトリー, 東京都
 Denso IT Laboratory, Inc., Shibuya CROSS TOWER 28th
 Floor, 2-15-1 Shibuya, Shibuya-ku, Tokyo, 150-0002 Japan

a) E-mail: tkhr@vision.cs.chubu.ac.jp

b) E-mail: yuu@vision.cs.chubu.ac.jp

c) E-mail: manbai@d-itlab.co.jp

d) E-mail: yyoshida@d-itlab.co.jp

e) E-mail: takayoshi@cs.chubu.ac.jp

f) E-mail: hf@cs.chubu.ac.jp

DOI:10.14923/transinfj.2014IUP0004

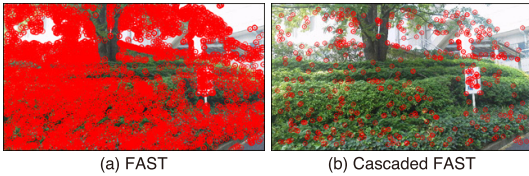


図 1 自然画像におけるコーナー検出結果の比較
Fig. 1 Comparison of corner detection results for a natural image.

用いることで効率的に周囲の画素を参照するため、高速にコーナーを検出できる。

しかし、FAST では図 1 に示すようなテクスチャが複雑な自然領域（木の葉、植え込み等）から多くのコーナーを検出する問題がある。これは、わずか周囲長 16 画素の輝度情報に基づいてコーナーを判定しているためである。

図 1(a) のように多くのコーナー点に基づき 2 画像間を対応付ける場合、二つの問題が発生する。一つ目の問題は、自然領域では視点の変化や、風による葉の揺らぎのような外乱の影響により見えの変化が生じやすいため、複数の画像間で同じコーナーを検出できないことがある。二つ目の問題は、2 画像間のコーナーの対応付けには、1 枚目から検出された一つのコーナーに対して 2 枚目から検出された全てのコーナーと比較する。したがって、検出されるコーナー数が多い場合には対応付けに多くの計算コストが必要となる。例として車載カメラや携帯電話端末による物体認識の事例を考える。このような事例の場合、背景に映り込んだ自然領域から多くのコーナー点を検出してしまい、計算コストが非常に高くなるおそれがある。このように認識対象物体以外の自然領域から検出される多くのコーナー点は処理速度の低下を招くため、検出されるコーナー点を抑制する必要がある。

そこで、本論文では画像間の対応付けに必要な点のみを高速に検出する Cascaded FAST を提案する。提案手法では FAST で参照する周囲長 16 画素に加え、より広範囲の画素を参照する決定木を用いる。そして、決定木をカスケード状に並べることで高速にコーナーを検出する。その結果として、提案手法は図 1(b) に示すように自然領域からのコーナー点の検出を抑制することができる。また、提案手法では画像ピラミッドからコーナーを検出することでスケールを得る。更に、周囲の画素を参照する提案手法の枠組みを利用することで、コーナー検出と同時にオリエンテーション

も得る。

1.1 関連研究

コーナー検出の研究は、コーナーの位置の正確さ (Localization) と 2 画像間におけるコーナーの再現率 (Repeatability) が重要とされ [10]、現在までに多くの研究 [1], [2], [7]~[9] がされている。コーナー検出の初期の研究として Moravec のコーナー検出 [7] が知られている。この方法は、注目画素を中心とする局所領域をパッチとし、コーナー点を中心とするパッチと近傍領域を中心とするパッチの見えの類似度が低いと仮定した。そして、類似度としてパッチの差分の 2 乗和によって求め、近傍領域で類似度が小さくなる点をコーナーとして検出した。Smallest Univalued Segment Assimilating Nucleus (SUSAN) [9] では、Moravec の 8 方向の輝度の変化強度しか考慮していないという欠点を円形マスクを用いることにより改善している。

一方、コーナー点はエッジの交点とも定義することができ、ある局所領域において x 方向と y 方向の二つの異なる方向のエッジが共に強いとき、その局所領域の中心はコーナーとみなせる。Harris [1] らは、このようなコーナー点を検出するために x , y 方向の局所的な勾配を要素にもつ行列の固有値の関係からコーナーを検出する。更に、Harris のコーナー検出をベースに、スケール変化やアフィン変形に対して頑健なコーナーを検出する Harris-Affine 検出器 [4] も提案されている。

以上のコーナー検出法は、冒頭で述べた位置の正確さと再現性が高い一方で、高速にコーナーを検出できない課題は長らく解決されなかった。この課題に対して Rosten [2] らは、機械学習を導入することでコーナーを高速に検出する Features from Accelerated Segment Test (FAST) を提案した。FAST では注目画素と注目画素を中心とする周囲長 16 画素を比較し、周囲長 16 画素の輝度値が注目画素より明るい、または暗い画素が連続する場合に注目画素をコーナーとして定義する。そして、このコーナーの定義に従い ID3 [11] のアルゴリズムにより学習した決定木をコーナー検出器とする。FAST は検出速度と性能が優れていることから、FAST をベースに改良した手法も幾つか提案されている [12], [13]。しかしながら、FAST は注目画素周辺のわずか 16 画素に基づいてコーナーを定義しているため、テクスチャが複雑な領域から多数のコーナーを検出する問題がある。

また、多くのコーナー検出法では、検出したコー

ナーの座標のみを出力する．しかし，画像のスケール変化や回転が加わった場合には，2 画像間の対応付けはできない．そこで [3] では，画像にスケール変化や回転を加えた場合においても同一の特徴量を得るために，特徴点の座標以外に基準となるスケールとオリエンテーションを獲得している．本論文では，座標とスケール，オリエンテーションをもつ点をキーポイントと呼ぶ．

1.2 提案手法の概要

提案手法は下記の三つの特徴をもつ．

- 複雑なテクスチャ領域でのコーナー検出の抑制
FAST では注目画素を中心としたわずかに周囲長 16 画素の同心円の輝度情報のみを用いてコーナーを定義するため，複雑なテクスチャをもつ領域から多数のコーナーを検出する．この問題を解決するために，提案手法ではより広範囲の画素を参照することでコーナーを定義する．

- 高速なコーナー検出

各注目画素において周囲長が {20, 16, 12} 画素の同心円を観測する三つの決定木を学習し，カスケード状に並べる．その結果，非コーナー点を早期に棄却できるため，高速なコーナー検出が実現できる．

- スケールとオリエンテーションの獲得

FAST では，コーナーの座標のみを出力する．一方，提案手法は，画像ピラミッドからコーナーを検出することでスケールを得る．更に，同心円上の画素を参照する提案手法の枠組みを利用することで，高速にオリエンテーションを得る．これにより，画像のスケール変化や回転に対して頑健なキーポイントとなる．

2. Features from Accelerated Segment Test (FAST) と問題点

機械学習を導入することで高速にコーナーを検出する FAST について述べる．

2.1 FAST によるコーナー検出

FAST は高速にコーナーを検出するために，機械学習により学習した決定木を用いてコーナーを検出する．

まず，決定木の学習に必要なコーナーと非コーナーのラベルを定義する．学習画像の全ての画素（学習サンプル）において式 (1) のように，注目画素 p を中心とする周囲長 16 画素を明るい (Brighter), 類似 (Similar), 暗い (Darker) の 3 値に分類する．

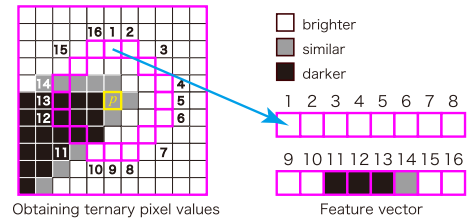


図 2 注目画素 p の特徴ベクトル

Fig. 2 Feature vector for pixel of interest p .

$$S_{p \rightarrow x} = \begin{cases} \text{Brighter} & I_p + t \leq I_{p \rightarrow x} \\ \text{Similar} & I_p - t < I_{p \rightarrow x} < I_p + t \\ \text{Darker} & I_{p \rightarrow x} \leq I_p - t \end{cases} \quad (1)$$

ここで， I_p は注目画素の輝度値， $x \in \{1, \dots, 16\}$ は円周上の画素の位置， $I_{p \rightarrow x}$ は周囲長 16 画素の輝度値， t はしきい値を表す．このように 3 値化した周囲長 16 画素を図 2 に示すように特徴ベクトルとする．そして，各注目画素 p のラベルは周囲長 16 画素のうち n 画素以上連続して Brighter 若しくは Darker のときコーナーとし，条件を満たさない場合は非コーナーと定義する．

次に，3 値化した周囲長 16 画素と注目画素のコーナーのラベルを用いて ID3 のアルゴリズムに従って決定木を学習する．分岐ノードでは円周上の画素の値 $S_{p \rightarrow x}$ を観測し，式 (2) で求められる情報利得 G が最も高い円周上の画素 x を用いて分岐ノードに辿り着いた学習サンプルを分岐する．

$$G = H(P) - H(P_b) - H(P_s) - H(P_d) \quad (2)$$

ここで， P は分岐ノードに辿り着いた学習サンプルの集合， P_b は $S_{p \rightarrow x} = \text{Brighter}$ と判定された学習サンプルの集合， P_s は $S_{p \rightarrow x} = \text{Similar}$ と判定された学習サンプルの集合， P_d は $S_{p \rightarrow x} = \text{Darker}$ と判定された学習サンプルの集合である． H はエントロピーを表し，式 (3) より求められる．

$$H(P) = (C + \bar{C}) \log_2(C + \bar{C}) - C \log_2 C - \bar{C} \log_2 \bar{C} \quad (3)$$

ここで， C はコーナー数， $\bar{C}$ は非コーナー数を示す． $H(P_b), H(P_s), H(P_d)$ においても $H(P)$ と同様に，分岐ノードに辿り着いた学習サンプル集合に含まれるコーナー数 C と非コーナー数 $\bar{C}$ を用いてエントロピーを計算する．この処理をコーナーと非コーナーの分類ができなくなるまで，すなわち情報利得が 0 にな

るまでノードを分岐する。情報利得が 0 となったときのノードを末端ノードとし、辿り着いたコーナーのラベルを記録する。

決定木によりコーナーを検出する際には、図 3 に示すように機械学習により学習した決定木へ注目画素 p を入力し、トラバーサルする。そして、到達した末端ノードに記録されたラベル情報によりコーナー若しくは非コーナーと判定する。決定木を用いることで周囲長 16 画素のうち平均で 2.26 画素を観測するだけでコーナーを検出することができるため、高速なコーナー検出が可能である。

2.2 FAST の問題点

高速にコーナーを検出できる一方、図 1 のようなテクスチャが複雑な自然領域において多数のコーナーを検出する問題がある。このような点は視点の変化や風による葉の揺らぎにより見えが容易に変化するため、複数の画像間で同じコーナーを検出できないことがある。また、多数のコーナーを検出すると 2 画像間の対応付けに必要な特徴量記述と距離計算の計算コストが大きくなる問題がある。したがって、テクスチャが複雑な自然領域から検出されるコーナーらしくない点を抑制する手法が必要である。

2.3 コーナー点の傾向調査

FAST によって検出されるコーナーらしい点とコーナーらしくない点の傾向を調査する。まず、コーナー周辺領域の見えの傾向調査をする。見えの傾向調査では、コーナーとして検出された点を中心とするパッチ画像を生成する。このパッチ画像の見えをコーナーらしい点とコーナーらしくない点において比較する。図 4 は、人工物画像と自然画像から FAST により検出したコーナーを示す。図 4(a) のコーナーらしい点は、どのパッチの見えも類似している傾向が確認できる。一方、図 4(b) のコーナーらしくない点は、見えのばら

つきが大きい傾向がある。

図 4 のコーナーの見えの傾向から、コーナーらしい点では周囲長 16 画素の外側と内側の周囲の画素も同じような輝度の変化であることが予想できる。そこで、コーナーを中心とする周囲の画素と注目画素の輝度の差分値の傾向を定量的に解析する。解析対象である周囲の画素を図 5 に示す。シアン、マゼンタ、青、赤、緑、オレンジの各色で示す領域はそれぞれ周囲長 {32, 28, 20, 16, 12, 8} 画素の同心円である。解析には人工物のみを含む画像と自然領域のみを含む画像から検出された 1,000 点のコーナーを使用し、差分値を解析する。その際、 n 画素以上の Brighter 若しくは Darker が連続する画素の始点を基準点とし、注目画素と基準点を結ぶ直線を基準線（角度 0° ）とする。図 6 に解析

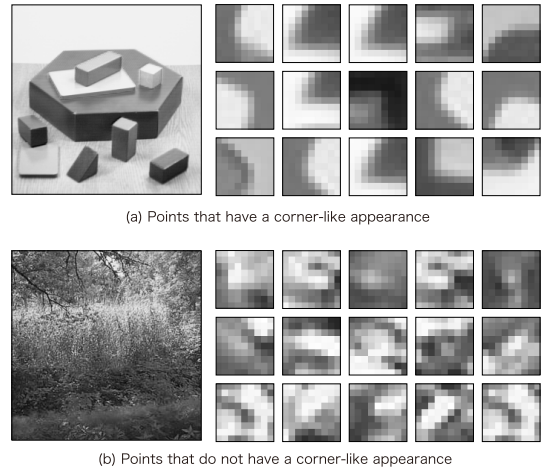


図 4 FAST により検出されたコーナーの見えの違い
Fig. 4 Differences in appearance of corners detected by FAST.

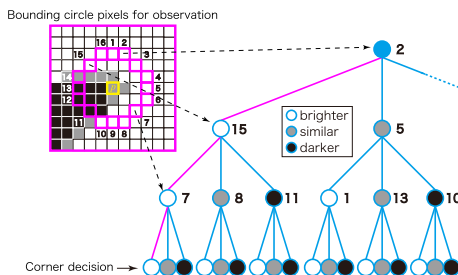


図 3 決定木を用いたコーナー検出
Fig. 3 Corner detection with a decision tree.

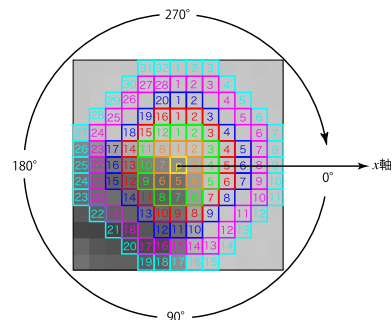


図 5 調査対称の領域
Fig. 5 A region being examined.

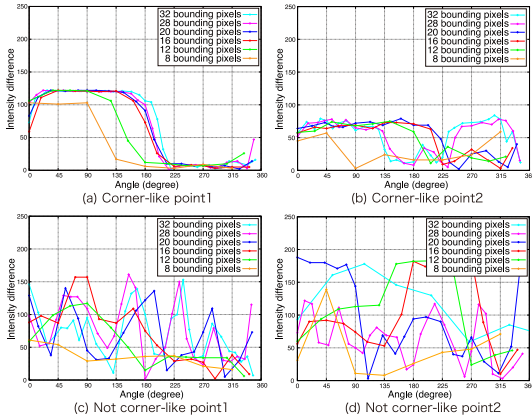


図 6 コーナー点の解析結果
Fig. 6 Results of corner point analysis.

結果を示す。図 6 のグラフの縦軸は注目画素と周囲の画素の差分の絶対値、横軸は角度を表す。コーナーらしい点の周囲長 {32, 28, 20, 16, 12} 画素の同心円の差分値は大きな値が連続し、グラフの形状が類似していることが確認できる。コーナーらしくない点では周囲長 {32, 28, 20, 16, 12, 8} 画素の同心円の差分値にばらつきがある。この解析結果から、周囲長 {32, 28, 20, 16, 12} 画素の同心円の情報が有効であると考えられる。しかし、周囲長 {32, 28} 画素の同心円はある点においては図 6(a) のように類似したグラフ形状が得られるが、同心円の半径が大きいため、図 6(b) のように大きな差分値が連続しない場合がある。また、周囲長 {32, 28} 画素の同心円のように同心円のサイズが大きくなると画像ピラミッドによりキーポイントのスケールを得る場合に小さな画像からキーポイントが検出されなくなるおそれがある。よって、提案手法では周囲長 {20, 16, 12} 画素の同心円の情報に基づきコーナーを検出する。

3. Cascaded FAST によるキーポイント検出

2.3 の調査結果から、コーナーらしい点の周囲長 {20, 16, 12} 画素の同心円において差分値の変化の傾向が似ていることを確認した。そこで、本研究では周囲長 {20, 16, 12} 画素の同心円の情報に基づいてコーナーを検出する。また、Cascaded FAST ではコーナーの座標に加え、スケールとオリエンテーションも求める。

3.1 コーナーの定義

Cascaded FAST では周囲長 {20, 16, 12} 画素の同心円において Brighter または Darker が連続しているかという条件に基づきコーナー候補点を検出する。そして、検出したコーナー候補点に対してオリエンテーションを算出し、周囲長 {20, 16, 12} 画素の同心円のオリエンテーションが類似している場合に注目画素をコーナーとして検出する。以下に各処理の詳細について述べる。

Step1: Brighter または Darker の連続性による条件

周囲長 {20, 16, 12} 画素の同心円上の画素を式 (1) により Brighter, Similar, Darker の 3 値に分類する。FAST では周囲長 16 画素において Brighter または Darker が 9 画素以上連続した場合に注目画素をコーナーとするが、提案手法では周囲長 {20, 16, 12} 画素の同心円上の画素がそれぞれ {11, 9, 6} 画素以上の Brighter または Darker が連続している場合に注目画素をコーナー候補点とする。周囲長 {20, 16, 12} 画素の同心円における Brighter または Darker の連続する画素数は、ベースとなる FAST の周囲長 16 画素の比率に合わせて決定した。

Step2: オリエンテーションの算出

Step1 で求めたコーナー候補点に対して周囲長 {20, 16, 12} 画素の同心円のそれぞれのオリエンテーションを算出する。図 7 に周囲長 16 画素のオリエンテーションの算出する例を示す。まず、Brighter または Darker が連続する画素の始点から終点までの角度を求める。 x 軸に対する注目画素 $p(u_p, v_p)$ と始点の画素 $x_s(u_s, v_s)$ の角度を θ_s 、 x 軸に対する注目画素 $p(u_p, v_p)$ と終点の画素 $x_e(u_e, v_e)$ の角度を θ_e とすると、始点と終点の角度 $\theta_{s \rightarrow e}$ は次式により求められる。

$$\theta_{s \rightarrow e} = \begin{cases} 360 - |\theta_s - \theta_e| & \text{If } \theta_s > \theta_e \\ |\theta_s - \theta_e| & \text{Otherwise} \end{cases} \quad (4)$$

$$\theta_s = \text{angle}(x_s, p) \quad (5)$$

$$\theta_e = \text{angle}(x_e, p) \quad (6)$$

ここで、 $\text{angle}(\cdot)$ は x 軸を基準とした始点または終点の角度を返す関数である。そして、始点と終点の角度を 2 等分する方向をオリエンテーション θ として次式より算出する。

$$\theta = \frac{\theta_{s \rightarrow e}}{2} + \theta_s \quad (7)$$

周囲長 {20, 12} 画素のオリエンテーションも図 7 と同様に算出する。

Step3:オリエンテーションの類似性による条件

検出されたコーナー候補点がコーナーらしい場合、周囲長 {20, 16, 12} 画素の同心円の Brighter または Darker の連続画素は整合性があると考えられる。この整合性を見るために各同心円のオリエンテーションを求め、各オリエンテーションの類似性によりコーナーと非コーナーを判定する。図 8 に示すように周囲長 16 画素の同心円のオリエンテーションと周囲長 12 画素の同心円のオリエンテーションの角度差を α 、周囲長 16 画素の同心円のオリエンテーションと周囲長 20 画素の同心円のオリエンテーションの角度差を β として求める。そして α と β が式 (8) を満たす場合にコーナー候補点である注目画素 $p(u_p, v_p)$ をコーナーとして検出する。

$$p(u_p, v_p) = \begin{cases} C & \text{If } \alpha \leq Th_1 \text{ \& } \beta \leq Th_2 \\ \bar{C} & \text{Otherwise} \end{cases} \quad (8)$$

ここで、 Th_1 と Th_2 はしきい値を表す。周囲長 20 画素のオリエンテーションと周囲長 12 画素のオリエ

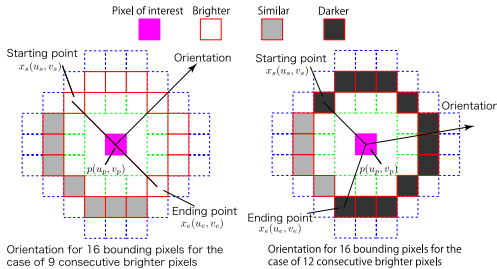


図 7 周囲長 16 画素におけるオリエンテーション算出例
Fig.7 Example of calculating orientation for 16 bounding pixels.

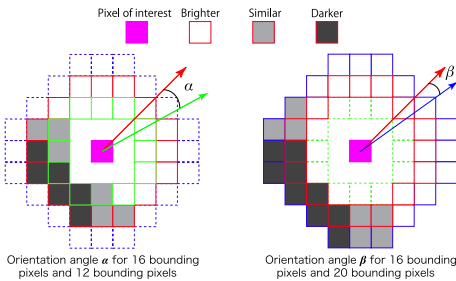


図 8 角度差 α と β の算出例

Fig. 8 Example of calculating angle differences α and β .

ンテーションはそれぞれ分解能が異なるため、 α と β に対して別々のしきい値を用いる。

3.2 機械学習による決定木の学習

提案手法においても FAST と同様に機械学習を導入することで高速なコーナー検出が可能である。決定木は周囲長 {20, 16, 12} 画素の同心円から一つの決定木を学習する方法と周囲長 {20, 16, 12} 画素の同心円を参照する三つの決定木をそれぞれ学習する方法が考えられる。前者の学習方法では参照画素数が増えることにより決定木の階層が深くなり処理速度の低下を招くおそれがある。一方、後者の学習方法では参照画素数の少ない決定木を組み合わせることにより、非コーナーを効率的に棄却することができるため、周囲長 {20, 16, 12} 画素の同心円ごとに決定木を学習する。

まず 3.1 により学習サンプルをコーナーと非コーナーに分類する。そして、分類した学習サンプルを ID3 のアルゴリズムに基づき、それぞれ周囲長 {20, 16, 12} 画素の同心円を参照する三つの決定木を学習する。

3.3 カスケード化による高速化

提案手法ではある画素を入力した際に、3.2 で学習した三つの決定木が全てコーナーと判定した場合にコーナー候補点と出力する。もし、一つの決定木でも非コーナーと判定した場合には非コーナーとして出力する。これをより高速に処理するために、それぞれの決定木をカスケードに配置する。これにより、各決定木で非コーナーとして判定された場合は早期に棄却できるため、より高速にコーナー候補点を検出することが可能となる。図 9 に決定木のカスケード化を示す。そして、検出されたコーナー候補点に対して Step2 と同様にオリエンテーションを算出し、Step3 にて述べたオリエンテーションの類似性を測ることでコーナーを判定する。周囲長 {20, 16, 12} 画素の同心円を参照する決定木の配置順は、コーナー検出結果に一切影響を与えない。しかし、各決定木の非コーナーの棄却率

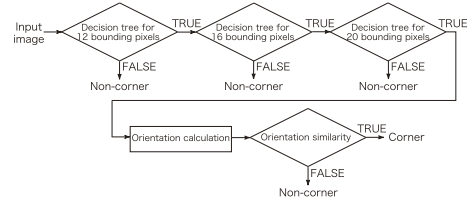


図 9 Cascaded FAST によるコーナー検出の流れ
Fig.9 Corner detection flow in Cascaded FAST.

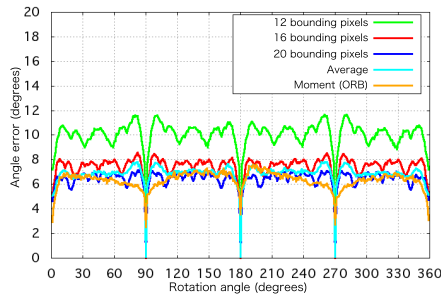


図 10 オリエンテーションの誤差
Fig. 10 Orientation error.

と同心円の参照画素数の関係から決定木の配置順により検出時間に違いが表れることを事前実験により確認した．図 9 に示すように，周囲長 {12, 16, 20} 画素の同心円を参照する決定木の配置順が最も高速にコーナーを検出できることを確認した．各決定木において非コーナーの棄却率はほぼ同等である．しかし，同心円上の参照画素数は各決定木において異なり，参照画素数が少ないほど決定木の実行速度が速いため，図 9 に示す配置順が最も高速であったといえる．

3.4 オリエンテーションとスケールの獲得

ここまで述べてきた手法では，コーナー点の座標のみを出力するコーナー検出法である．提案する Cascaded FAST は，コーナー点の座標に加え，スケールとオリエンテーションも出力することが可能である．スケールについては，複数の解像度で表現される画像ピラミッドから Cascaded FAST によりコーナー点を検出することで，検出した際に入力された画像解像度のスケールを利用する．複数のスケールで同じ位置のコーナーが検出された場合は別々のキーポイントとして検出する．オリエンテーションについては，3.1 の Step2 により算出されるオリエンテーションを出力する．図 10 は，画像を 1° ～ 359° まで回転させたときのオリエンテーションの平均誤差を表した結果である．例えば，画像を 45° 回転させた場合，元画像と回転画像で対応する点のオリエンテーションの角度を差分し，差分値が 45° であるとき誤差は 0 となる．この方法で周囲長 {20, 16, 12} 画素の同心円のオリエンテーション，周囲長 {20, 16, 12} 画素の同心円のオリエンテーションの平均，ORB で用いられるモーメントに基づいて算出したオリエンテーション [14] を比較した．

各周囲長の同心円のオリエンテーションは始点と終点の位置のずれと Brighter または Darker の連続画素

表 1 各同心円の最大・最小誤差
Table 1 Maximum and minimum error of each bounding pixels.

	周囲長 12 画素	周囲長 16 画素	周囲長 20 画素
最小誤差	13.28°	9.22°	7.02°
最大誤差	36.87°	26.57°	22.62°

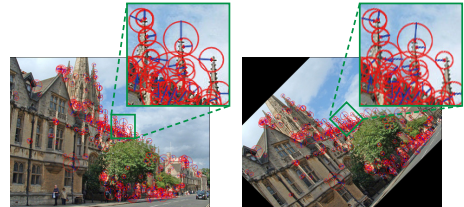


図 11 Cascaded FAST によるキーポイント検出例
Fig. 11 Example results of keypoint detection by Cascaded FAST.

数のずれにより誤差が発生すると考えられる．各周囲長の同心円において始点・終点が 1 画素ずれた場合と Brighter または Darker の連続画素数が 1 画素ずれた場合の誤差を表 1 に示す．この結果，図 10 の各周囲長の同心円のオリエンテーションの平均誤差は各同心円の最大誤差より低いため妥当な数値であると考えられる．また，提案手法である Cascaded FAST の同心円ごとのオリエンテーションの誤差を見ると，周囲長 20 画素を参照する決定木のオリエンテーションの誤差が小さいことが確認できる．提案手法では，周囲長 20 画素を参照する決定木のオリエンテーションを採用する．また，モーメントに基づくオリエンテーションと周囲長 20 画素を参照する決定木のオリエンテーションの精度が同等であることが分かる．図 11 に Cascaded FAST により検出されたキーポイントを示す．赤色の円の中心はコーナーの座標，円の大きさはスケールの大きさ，青色の線はオリエンテーションを表す．図 11 から画像を回転させた場合，検出されたコーナーのオリエンテーションも同様に回転していることが確認できる．

4. 評価実験

Cascaded FAST の有効性を確認するために評価実験を行う．本章では，コーナー検出時間の評価，F 値によるコーナー検出の評価，2 画像間の特徴点の対応付けの評価に分けて実験する．各実験では，Harris, FAST, Cascaded FAST のコーナー検出法を比較する．Harris のコーナー検出法は OpenCV により

表 2 各手法の検出時間の比較

Table 2 Comparison of detection times for various methods.

	Harris	FAST	Cascaded FAST
検出時間 [ms]	164.4	4.5	7.4
コーナー数	1134	8580	1197

実装されたコード^(注1)を使用し、FAST のコーナー検出法は著者らにより公開されているコード^(注2)を用いる。また、各実験では CPU: Intel(R) Xeon(R) X5470 3.33GHz, メモリ: 32GB の PC を用いる。

4.1 コーナーの検出時間の評価

ここでは Harris, FAST, Cascaded FAST のコーナー検出時間を比較する。本実験では各手法のコーナー検出時間を比較するため、スケールとオリエンテーションは求めない。画像は [15] のデータセットから人工物と自然領域を含む画像 (1024 × 768 画素) を 1,000 枚を用いる。

表 2 に各手法の比較結果を示す。表 2 は各手法において画像 1 枚当たりの平均処理時間と検出した平均コーナー数を示している。Cascaded FAST の処理時間は FAST に比べて 2.9[ms] 増加するが、Harris と比較して約 22 倍高速にコーナーを検出できる。また、Cascaded FAST は約 135[fps] で動作可能である。

4.2 F 値による評価

ここでは、コーナーらしくない点の検出を抑制できているかを確認するため、Cascaded FAST と FAST により検出されたコーナーの F 値を比較する。また、Cascaded FAST のオリエンテーションの類似性による条件の効果を確認するため、オリエンテーションによる判定をしない場合の手法を比較する。データセットには 4.1 と同じ画像を用いる。抑制するべき点とそうでない点を正確に定義することができないため、本研究では Harris により検出されたコーナー点を真のコーナーとして、Cascaded FAST と FAST の F 値を評価に用いる。

Harris のコーナー検出法は x 方向の勾配 I_x , y 方向の勾配 I_y を要素にもつ行列 $\mathbf{M}$ を用いてコーナーらしさを求める。コーナー検出時には式 (10) のレスポンス値 R をしきい値処理することによりコーナーを判定する。適切なしきい値を用いることで図 12 のように自然領域から検出されるコーナーを抑制し、信頼性の高いコーナーを検出することができる。



図 12 Harris のコーナー検出結果
Fig. 12 Corner detection result of Harris.

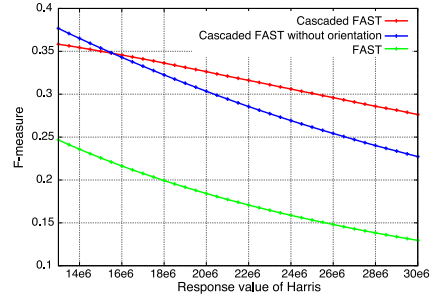


図 13 Cascaded FAST と FAST の F 値
Fig. 13 F-measure of Cascaded FAST and FAST.

$$\mathbf{M} = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (9)$$

$$R = \det \mathbf{M} - k \cdot \text{trace} \mathbf{M}^2 \quad (10)$$

図 13 に各手法の F 値の結果を示す。グラフの縦軸に F 値を示し、横軸は Harris のレスポンス値 R のしきい値を示す。図 13 の結果から Cascaded FAST は FAST より F 値が高いことが確認できるため、コーナーらしい点のみを検出できていると考えられる。また、Cascaded FAST はオリエンテーションの類似性による判定をしない場合、Harris のレスポンス値のしきい値を高くするほど F 値の差が大きくなる。レスポンス値のしきい値が高い場合、コーナー点の信頼性が高いといえるため、オリエンテーションの類似性によりコーナーを検出することは有効であるといえる。

4.3 2 画像間の対応付けにおける性能と速度

ここでは、2 画像間のキーポイントのマッチング性能と時間を評価する。比較手法を表 3 に示す。スケールの算出には、どの手法においても画像ピラミッドを用いる。オリエンテーションの算出は、[14] で用いられるコーナー点を中心とするパッチ内のモーメントに基づく手法と Cascaded FAST の 2 種類である。特徴量は、回転とスケールの変化を正規化した ORB 特徴量 [14] を用いる。ORB 特徴量はパッチ内の輝度の

(注1) : <http://opencv.org>

(注2) : <http://www.edwardrosten.com/work/fast.html>

小関係を 0 と 1 のバイナリコードで表現した特徴量である。本実験では、4.1 で用いた画像 130 枚に対して 10 種類のアフィン変換を施した 1,300 枚の画像を使用する。これにより変換前後の座標位置の同定が容易にできる。2 画像間のキーポイントを対応付けるためには、1 枚目の画像から得られる一つのキーポイントと 2 枚目の画像から得られる全てのキーポイントとの距離を計算し、最も距離が小さい点 p_1 と 2 番目に距離が小さい点 p_2 を求める。そして式 (11) が成立する場合にキーポイントを対応付ける。

$$Dist(p_1) < Dist(p_2) \cdot T \quad (11)$$

ここで、 $Dist(\cdot)$ は距離関数、 T はしきい値を表す。また、本実験ではバイナリコードで表現される特徴量を用いるため、ハミング距離により高速な距離計算が可能である。式 (11) が成立する場合、2 点の座標が同

一であるかを判定する。座標が $\sqrt{(1+1)}$ 画素以内の位置ずれの場合に正解とする。キーポイントの対応付け性能の評価は、マッチング率 (= 正解数/対応付け数) を用いる。そして、マッチング率とフレームレート (fps) を比較する。

各手法のマッチングの性能と速度の結果を図 14 に示す。提案する Cascaded FAST は他の手法と比較して、マッチング率はほぼ同等で、フレームレートは高いことが確認できる。図 15 に 2 画像間の対応付けの各処理の計算時間の内訳を示す。Harris のコーナー検出は、コーナー検出の処理時間が占める割合が非常に大きい。一方、FAST はコーナー検出とオリエンテーション算出を高速に処理できるが、特徴量記述と距離計算に多大な計算時間を要する。この理由として、FAST は他の手法と比べて非常に多くのコーナーを検出するためである。提案手法である Cascaded FAST は、他

表 3 各手法のオリエンテーション、スケール、特徴量
Table 3 Methods for obtaining orientation, scale, and descriptors.

手法	オリエンテーション算出	スケール取得	特徴量記述
Harris	モーメント	画像ピラミッド	ORB
FAST	モーメント	画像ピラミッド	ORB
Cascaded FAST(モーメント)	モーメント	画像ピラミッド	ORB
Cascaded FAST(周囲長 20 画素)	Cascaded FAST	画像ピラミッド	ORB

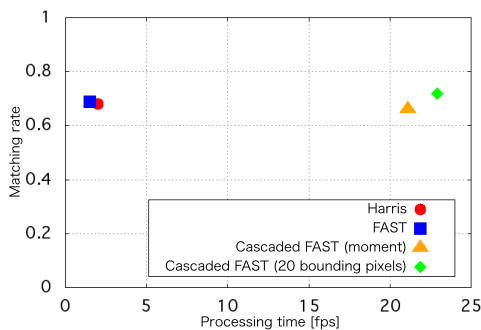


図 14 マッチングの性能と速度
Fig. 14 Matching performance and speed.

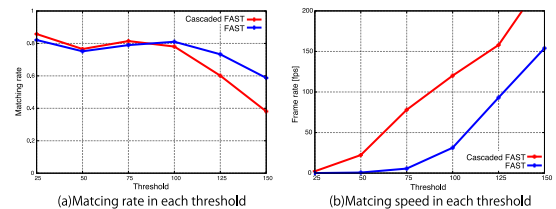


図 16 各しきい値における Cascaded FAST と FAST の性能
Fig. 16 Performance of C-FAST and FAST in each threshold.

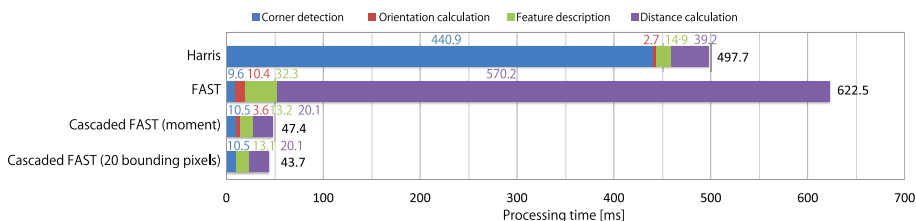


図 15 2 画像間の対応付けの各処理の計算時間の内訳
Fig. 15 Breakdown of processing computation time for keypoint matching.

の2手法と比べて高速に各処理を実行できていることが分かる。特に、周囲長20画素の情報を用いてオリエンテーションを算出する場合は、コーナー検出時に計算したオリエンテーションを再利用することができるため、この処理を省くことができる。周囲長20画素の情報を用いてオリエンテーションを用いたCascaded FASTは、2画像間の対応付けを約43.7[ms]で処理することが可能である。また、Cacaded FASTとFASTは図16に示すように式(1)のしきい値により性能が変化するため、しきい値を変化させたときのマッチング率とフレームレートを比較する。図16(a)はしきい値を変化させたときのマッチング率を示す。しきい値が25から100の範囲では同等の性能を維持しているが、しきい値を高く設定すると検出されるコーナー点数が極端に少なくなるため、マッチング性能が低下する。図16(b)はしきい値を変化させたときのフレームレートを比較した結果である。しきい値を低く設定すると検出されるコーナー数が多くなるため、フレームレートが低くなり、しきい値を高く設定すると検出されるコーナー数が少なくなるため、フレームレートが高くなる。図16(a), (b)の結果から性能を維持しつつ高速な処理を行うためには式(1)のしきい値 t は50から100の範囲が妥当であると考えられる。

5. む す び

本論文では自然領域からのコーナー点の検出を抑制することで高速に2画像間に対応づけるCascaded FASTを提案した。提案手法では、周囲長{20, 16, 12}画素の3種類の同心円の輝度値の連続性とオリエンテーションの類似性によりコーナーを定義することで自然領域からのコーナー検出を抑制した。また、FASTのアプローチで学習した三つの決定木をカスケード状に並べて非コーナーを早期に棄却することで高速なコーナー検出を実現した。更に、提案手法では画像ピラミッドを用いることでスケールを獲得し、周囲の画素を参照する枠組みを利用することで高速にオリエンテーションを獲得することができ、対応点マッチングのキーポイントとして有効である。今後の課題は、画像ピラミッドから得られるスケールを効率的に獲得することである。

文 献

[1] C. Harris and M. Stephens, "A combined corner and edge detector," *Alvey Vision Conference*, pp.147–151, 1988.

- [2] E. Rosten, R. Porter, and T. Drummond, "FASTER and better: A machine learning approach to corner detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.32, pp.105–119, 2010.
- [3] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol.60, pp.91–110, 2004.
- [4] Mikolajczyk and Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol.60, no.1, pp.63–86, 2004.
- [5] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," *British Machine Vision Conference*, pp.36.1–36.10, 2002.
- [6] T. Tuytelaars and K. Mikolajczyk, *Local Invariant Feature Detectors: A Survey*, Now Publishers Inc., Hanover, MA, USA, 2008.
- [7] H. Moravec, "Rover visual obstacle avoidance," *International Joint Conference on Artificial Intelligence*, pp.785–790, 1981.
- [8] J. Shi and C. Tomasi, "Good features to track," *Conference on Computer Vision and Pattern Recognition*, pp.593–600, 1994.
- [9] S.M. Smith and J.M. Brady, "SUSAN—A new approach to low level image processing," *International Journal of Comput. Vision*, vol.23, no.1, pp.45–78, 1995.
- [10] J.P. Gravel, "Corner detection," *Biological Cybernetic*, vol.59, no.4, pp.139–153, 1988.
- [11] J.R. Quinlan, "Induction of decision trees," *Machine Learning*, vol.1, no.1, pp.81–106, 1986.
- [12] E. Mair, G.D. Hager, D. Burschka, M. Suppa, and G. Hirzinger, "Adaptive and generic corner detection based on the accelerated segment test," *Conference on Computer Vision*, pp.183–196, 2010.
- [13] J. Cronje, "BFROST: Binary features from robust orientation segment tests accelerated on the GPU," *Annual Symposium of the Pattern Recognition Association of South Africa*, pp.25–30, 2011.
- [14] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," *International Conference on Computer Vision*, pp.2564–2571, 2011.
- [15] "The Oxford Buildings Dataset," <http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/>, 2012.

(平成26年7月1日受付, 10月31日再受付,
27年1月9日早期公開)



長谷川 昂宏 (学生員)

2013 年中部大学工学部情報工学科卒業。同年より同大学大学院博士前期課程に在籍。画像処理、パターン認識に関する研究に従事。



山内 悠嗣 (正員)

2012 年中部大学大学院博士後期課程修了。2012 年中部大学院博士研究員，2014 年中部大学助手。2010 年独立行政法人日本学術振興会特別研究員 DC2。コンピュータビジョン、パターン認識の研究に従事。



安倍 満 (正員)

2007 年慶應義塾大学大学院博士後期課程修了。博士 (工学)。2007 年株式会社デンソーアイティーラボラトリ入社。2011 年画像センシングシンポジウム (SSII) 最優秀学術賞。パターン認識・理解、コンピュータビジョンの研究に従事。IEEE 会員。



吉田 悠一

2003 年大阪大学大学院修士課程修了。2003 年日本電信電話株式会社。2007 年株式会社デンソーアイティーラボラトリ入社。パターン認識・理解、コンピュータビジョン、ユーザインタフェースの研究に従事。



山下 隆義 (正員)

2002 年奈良先端科学技術大学院大学博士前期課程修了。2002 年オムロン株式会社入社。2009 年中部大学大学院博士後期課程修了 (社会人ドクター)。2014 年中部大学講師。人の理解に向けた動画画像処理、パターン認識・機械学習の研究に従事。2009 年画像センシングシンポジウム高木賞。2013 年電子情報通信学会情報・システムソサイエティ論文賞。2013 年電子情報通信学会 PRMU 研究会研究奨励賞。



藤吉 弘亘 (正員)

1997 年中部大学大学院博士後期課程修了。1997～2000 年米カーネギーメロン大学ロボット工学研究所 Postdoctoral Fellow。2000 年中部大学講師，2004 年同大准教授を経て 2010 年より同大教授。2005～2006 年米カーネギーメロン大学ロボット工学研究所客員研究員計算機視覚、動画画像処理、パターン認識・理解の研究に従事。2005 年ロボカップ研究賞。2009 年情報処理学会論文誌コンピュータビジョンとイメージメディア優秀論文賞，2009 年山下記念研究賞。2010 年，2013 年，2014 年画像センシングシンポジウム優秀学術賞。2013 年電子情報通信学会情報・システムソサイエティ論文賞。博士 (工学)。情報処理学会，IEEE 各会員。