

# Cascaded FASTによる高速なキーポイント検出

長谷川 昂宏\*, 山内 悠嗣, 藤吉 弘亘 (中部大学)

High-speed Keypoint Detection by Cascaded FAST

Takahiro Hasegawa, Yuji Yamauchi, Hironobu Fujiyoshi (Chubu University)

## 1. はじめに

高速なコーナー検出法として Features from Accelerated Segment Test (FAST) (1) が提案されている。しかし、FAST ではテクスチャが複雑な領域においてコーナーらしくない点を多く検出する問題がある。そこで、本研究では、参照する画素の異なる 3 種類の決定木をカスケード化したキーポイント検出法を提案する。

## 2. FAST によるコーナー検出と問題点

FAST は注目画素を中心とする周囲 16 画素の輝度と注目画素の輝度の差を閾値処理により Brighter, Similar, Darker の 3 値に分類する。そして 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して定義し、これを基に決定木を学習する。コーナー検出時は決定木をトラバーサルすることによりコーナーを検出できるため非常に高速である。

しかし、テクスチャが複雑な領域においてコーナーらしくない点を多く検出する問題がある。そこで、コーナーらしくない点とコーナーらしい点の周辺領域との差分値の傾向を調査する。Fig.1 に示すようにコーナーらしい点の周囲 {20,16,12} 画素の差分値は大きな値が連続し、同一の傾向である。しかし、コーナーらしくない点の差分値はばらつきがあることがわかる。

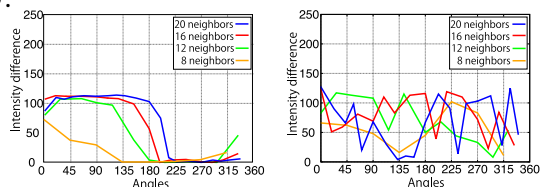


Fig. 1 Difference of tendencies between corner and non-corner.

## 3. Cascaded FAST

コーナー点の傾向調査に基づき、Fig.2 に示すように 3 段階の処理によりキーポイントを検出する。

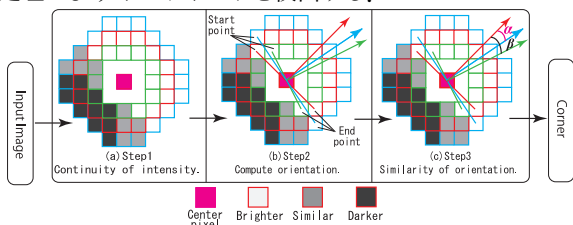


Fig. 2 Flow of the proposed method.

### Step1: Brighter または Darker の連続性による条件

周囲 {20, 16, 12} 画素を FAST と同様に Brighter, Similar, Darker に分類する (Fig.2(a)). Brighter または Darker がそれぞれ {11, 9, 6} 画素以上の場合に注目画素をキーポイント候補点とする。

### Step2: オリエンテーションの算出

キーポイント候補点に対して連続する Brighter または Darker の始点から終点までの角度を求める。その角度を 2 等分する

方向をオリエンテーションとする (Fig.2(b)).

### Step3: オリエンテーションの類似性による条件

周囲 {16, 12} 画素のオリエンテーション間の角度  $\alpha$ , 周囲 {16, 20} 画素のオリエンテーション間の角度  $\beta$  を求め、 $\alpha \leq Th_1$ ,  $\beta \leq Th_2$  の条件を満たす場合、注目画素をキーポイントとする (Fig.2(c)).  $Th_1$  と  $Th_2$  はそれぞれの閾値を示す。

### カスケード化による高速化

提案手法では、周囲 {20, 16, 12} 画素を参照する決定木を学習し、3つの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。出力するキーポイントは、コーナーの位置情報と Step2 により算出したオリエンテーションを持つ。また、画像ピラミッドを用いることで、スケールの出力も可能である。

## 4. 評価実験

評価実験により提案手法の有効性を示す。本研究では 2 画像間の対応付けにおけるマッチング率とフレームレートにより各手法を比較する。Fig.4 に示すように、マッチング率は各手法で同等であるが、処理時間を比較した場合、Cascaded FAST は Harris や FAST よりも高速に処理が可能である。FAST はコーナー検出においては非常に高速に処理できるが、自然領域から多くのコーナーを検出するため、2 画像間の対応付けでは多大な処理時間を要する。しかし、Cascaded FAST では不要なキーポイントの検出を抑制できるため、2 画像間の対応付けにおいても非常に高速である。



Fig. 3 Examples of keypoints detection.

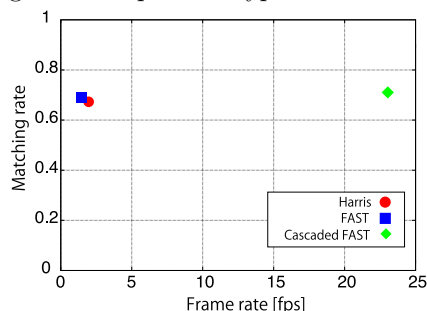


Fig. 4 Performance versus runtime.

## 5. おわりに

提案手法では周囲 {20,16,12} 画素を参照する決定木を用いることでコーナーらしい点のみを高速に検出することが可能である。今後は効率的なスケールの推定方法を検討する。

### 文献

(1) E. Rosten, *et al.*, "Machine learning for High-speed Corner Detection", ECCV, pp. 430-443, 2006.