

Brick Partitioning と初期閾値推定を用いたテンプレートマッチング法における
距離下限値初期分割レベルについての検討

松本 涼[†], 外山 史[‡], 森 博志[‡], 東海林 健二[‡]

宇都宮大学工学部[†]

宇都宮大学大学院工学研究科[‡]

1. はじめに

テンプレートマッチングは検索対象画像内から目的画像と最も類似度が高い位置を検出する技術である。最も基本的な手法として Full Search(FS)がある。FS は目的の画像(テンプレート)と検索対象画像上の各位置(ウィンドウ)の類似度(距離値)を逐次評価していき、最も類似度の高い位置を検出位置として出力する。この距離値には式(1)で定義される Sum of Absolute Difference(SAD)が用いられる。

$$SAD(x, y) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |T(i, j) - W(x + i, y + j)| \quad (1)$$

ここで、(x, y)は検索対象画像内におけるウィンドウの位置、T(i, j)及び W(i, j)はそれぞれテンプレート及び検索対象画像内の座標(i, j)における画素値であり、M, N はテンプレート画像のサイズである。FS は膨大な検索時間がかかるため、Multilevel SEA(MSEA) や Winner Update Algorithm(WUA)などの高速化手法が数多く提案されている。MSEA はテンプレートとウィンドウを段階的に分割して得られる距離下限値を用いて、絶対値距離の計算を削減し高速な処理を実現する手法である。また、WUA は初期段階ですべてのウィンドウに対して距離下限値を取得し、最小の距離下限値をもつウィンドウから距離計算を行っていく手法である。更に MSEA と WUA を組み合わせることで高速化を実現した手法として Brick Partitioning と初期閾値推定を用いた手法(EIT+BP : EffectiveInitial Thershold + Brick Partitioning)[1]がある。EIT+BP では、MSEA と WUA の利点を組み合わせ、更に新しい領域分割法を用いることにより高速化を実現している。EIT+BP では、最初にある規則で選択された少数のウィンドウから WUA を用いて最小となる SAD を計算する。この SAD を初期閾値 θ とし、残りのウィンドウを MSEA で探索することにより、効率のよいウィンドウのスキップを実現し

ている。閾値 θ は、現探索時点での SAD の最小値である。ここで、MSEA の探索処理において、すべてのウィンドウでは領域分割レベル(領域分割回数)が 0 から開始されている。しかし、ウィンドウの前後の画像は類似度が高いため、現ウィンドウでの領域分割レベルは、直前のウィンドウで得られた領域分割レベルに近い値となる。

そこで本研究では、EIT+BP の MSEA 探索処理において、直前のウィンドウで得られた領域分割レベルを元に、現ウィンドウでの領域分割の開始レベルを決定することにより、更に高速化を行う手法を提案する。

2. 従来手法 (EIT+BP)

2.1 概要

EIT+BP の全体の流れは以下の通りである。

1. すべてのウィンドウから等間隔にウィンドウを選択し WUA 探索を行う。
2. 1.の処理において、検出位置候補となったウィンドウ近傍の領域に対して WUA 探索を行う。
3. 2.で取得した最小 SAD を初期閾値として 1.,2.で探索を行っていないウィンドウを MSEA で探索する。

MSEA 探索において、ウィンドウは分割レベル 0 から始まり、閾値 θ を超えた段階でそのウィンドウにおける以降の距離計算をスキップする。最大分割レベルまで分割を行い、算出した距離下限値が閾値 θ を下回る場合は、そのウィンドウの SAD を計算し、それが閾値 θ を下回る場合はその値を新しい閾値 θ として更新する。EIT+BP の処理の具体的内容については文献[1]を参照されたい。

2.2 Multilevel SEA(MSEA)

MSEA はテンプレートとウィンドウを領域分割し、それぞれの部分領域間の画素の積算値の差の合計を距離下限値とし、この距離下限値が閾値 θ を上回る場合は以降の計算をスキップする手法である。また、算出した距離下限値が閾値 θ を下回る場合は、ウィンドウの領域を更に分割する(分割レベルを上げる)。分割レベルが最大のウィンドウにおける距離下限値が閾値 θ を下

Initial partition level for template matching using (brick partitioning and) initial threshold

[†]Ryo Matsumoto · Faculty of Engineering, Utsunomiya University

[‡]Toyama Fubito · [‡]Mori Hiroshi · [‡]Shoji Kenji

Graduated School of Engineering, Utsunomiya University

回る場合は、そのウィンドウの絶対値距離計算を行い、閾値 θ を更新し、次のウィンドウの計算を行っていく。L を分割レベルとすると、L=1 では $M/2 \times N/2$ の部分領域に分割される。これを $M/4 \times N/4$ を経て、 1×1 のサイズまで分割を繰り返す。ここで各分割レベルにおける領域分割の様子を図 1 に示す。 UB_L は分割レベル L の時の距離下限値を表しており、分割レベルの増加に伴い、距離下限値は単調増加を示す。

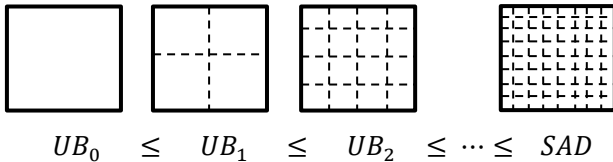


図 1 領域分解の様子

2.3 絶対値距離と距離下限値

距離下限値(UB_L)の計算は以下の式により求められる。

$$UB_L(x, y) = \sum_{p=0}^{K_L-1} \left| \sum_{i=0}^{P-1} T_{sub(i)}^p - \sum_{p=0}^{P-1} W_{sub(i)}^p \right| \quad (2)$$

式(2)の K_L は分割レベル L における部分領域の数を、P は部分領域に含まれる画素の合計値、 $sub(i)$ は分割された領域の i 番目を表す。 $T_{sub(i)}^p$, $W_{sub(i)}^p$ はそれぞれウィンドウ、テンプレートにおける i 番目の分割領域に含まれる p 番目の画素値である。式(2)における部分領域内の画素値の総和はインテグラルイメージによって高速に求めることができる。コーシー・シュワルツの三角不等式から以下の関係が成り立つ。

$$UB_0(x, y) \leq UB_1(x, y) \leq \dots \leq SAD(x, y) \quad (3)$$

式(3)より、距離下限値は分割レベルを増加させると、その値が大きくなるとともに SAD に近似的にいく。よって、分割レベルを上げていき、距離下限値が閾値 θ を超えた段階で、そのウィンドウをスキップすることが可能となる。

3. 提案手法

提案手法では EIT+BP の第 3 段階における MSEA 探索において、ウィンドウの分割レベルを直前のウィンドウの分割レベルより 1 低いレベルから開始する。これにより、従来手法より効率よく距離下限値の計算が行われ、高速な処理を実現することができる。図 2 に EIT+BP で探索を行った時のウィンドウの分割レベル分布の例を示す。図 2 はウィンドウの座標に割り当てられる分割レベル分布を視覚的に表したものである。分割レベルが高いほど白く、低いほど黒く表示



している。この画像より、隣接するウィンドウ間の分割レベルは関連性が高いことが分かる。EIT+BP では、すべてのウィンドウの分割レベルを 0 から開始しているが、隣接するウィンドウ間の分割レベルに関連性があることから、直前のウィンドウの分割レベルに近い値を初期分割レベルとすることで、無駄なレベルの距離下限値計算を削減することが可能となる。

4. 実験

実験では、提案手法の有効性を確認するため、従来手法(EIT+BP)と比較を行った。実験で用いたシステム及び開発環境は、CPU: Intel® Xeon® X5650 2.67GHz, Memory: 64GB である。実験では Caltech の画像データセットより選択した画像を検索対象画像として使用している。また、テンプレート画像には -30dB のガウシアンノイズを付加しているため、検出位置で求められる SAD は 0 より大きい値となる。EIT+BP との比較結果を表 1 に示す。

表 1. 従来手法との比較結果

	画像1	画像2	画像3
従来手法	0.0401	0.2129	0.909
提案手法	0.0373	0.2109	0.8992
改善率(%)	6.9656	0.939	1.0778

表 1 にはそれぞれの手法での計測時間(s)及び EIT+BP に対する改善率を示している。改善率は EIT+BP と提案手法の計測時間の差を EIT+BP の計測時間で割った値を 100 分率で示したものである。表 1 より、実験で使用したすべての画像に対して提案手法は高速な処理を実現していることが分かる。

5. おわりに

本研究では、EIT+BP の MSEA 探索処理において、ウィンドウの分割レベルを直前のウィンドウで得られた分割レベルより 1 低いレベルから開始することにより、高速化を行う手法を提案した。実験では、EIT+BP との比較を行うことで提案手法の有効性を示した。

参考文献

[1] 町井孝充, 外山史, 森博志, 東海林健二, “Blick Partitioning と初期しきい値推定を用いたテンプレートマッチングの高速化”, 電子情報通信学会論文誌 D, Vol.J1000-D, No.1, pp.93-103, 2017.