

PatchMatch を用いた類似パッチの高速 KNN 探索法 Fast KNN algorithm using PatchMatch

范 盈盈[†] 田中 正行[‡] 奥富 正敏[‡]
YingYing FAN Masayuki TANAKA Masatoshi OKUTOMI

1. まえがき

ある画像内の注目しているパッチ（小領域）に対し、類似度の高いパッチのことを類似パッチと呼ぶ。画像処理の分野では、パッチ間の類似度評価に画素値の差分距離を用いた SSD (Sum of Squared Difference) という指標が用いられることが一般的である。注目パッチと同一画像、あるいは異なる画像から、最も類似度の高いパッチを探索する方法は最近隣探索 (Nearest-Neighbor Searching) 法として様々な手法が提案されている。

注目パッチの類似パッチを探索する最も単純な方法として、画像全探索法が挙げられる。この手法は、注目パッチと、探索対象となる画像内の全てのパッチとの類似度評価値を計算することで、最も類似度評価値の高いパッチを探索する手法である。しかしながら、探索にかかる計算負荷が非常に大きい。

また、類似パッチを高速に探索する方法として、PatchMatch 探索法が提案されている[1]。PatchMatch 探索法は、近似的に類似パッチを探索する手法であり、全探索法に比べ、類似パッチの探索にかかる計算コストを大幅に削減することができる。

本論文では、PatchMatch 探索法を応用して、同一画像内から K 個の類似パッチ (K-Nearest Neighbor) を高速に探索する方法を提案する。提案手法では、パッチ間の評価に SSD のみならず、既に探索されたパッチとの空間的な距離も同時に考慮する。これにより、特定のパッチが何度も選ばれることなく、異なる K 個の類似パッチを高速に探索する。

さらに、本提案手法を、ノイズを含む一枚の画像からのノイズ除去に利用する。本手法を用いて探索した K 個のパッチを Non-Local Means フィルタ[2]に適用することでノイズ低減の効果を確認する。あわせて、ノイズ除去にかかる計算時間を算出し、全探索法と比べ、計算時間が大幅に短縮できることを示す。

2. PatchMatch による高速 KNN 探索

本章では、従来手法の PatchMatch の概要について述べる。

PatchMatch は、類似パッチを探索するためのランダム近似最近隣探索アルゴリズムである。一般的に画像の隣接パッチ間は相関が強いという性質がある。この特性を利用することで高速に探索を行うことができる。さらにランダムサーチを使用して探索領域を変化させることで、類似パッチの発見確率を高めている。

PatchMatch アルゴリズムは下記のステップで構成されている。

[†] 所属なし, No Affiliation

[‡] 東京工業大学大学院理工学研究科機械制御システム専攻, Graduate School of Science and Engineering, Tokyo Institute of Technology

①. 初期化: 注目パッチは、対応するパッチをランダムに与える。

②. 反復処理:

②-1. プロバゲーション: 注目パッチの隣接パッチに注目し、それぞれの対応パッチとの類似度評価値を算出する。

②-2. ランダムサーチ: 注目パッチとその隣接パッチの中で最も類似の領域の近傍をランダムサーチする。

上記の手順を行うことで、近似的に最も類似度の高いパッチを高速に探索する。

3. PatchMatch を用いた高速 KNN 探索

高速近似最近隣探索法の一つとして PatchMatch が有効であることはすでに述べたとおりである。本論文では、PatchMatch アルゴリズムを用いて、複数個の類似パッチを高速に探索するための方法について述べる。

3.1. 複数個の類似パッチを探索する際の問題

まず、最も類似度の高い類似パッチを 1 つ探索することを考える。本論文ではパッチ間の類似度評価として、式(1)の画素値の SSD を利用する。

$$SSD(\mathbf{p}, \mathbf{q}) = \sum_{i=0}^m \sum_{j=0}^m (p(i, j) - q(i, j))^2 \quad \dots \text{式 (1)}$$

ここで、 \mathbf{p} は注目パッチ、 \mathbf{q} は探索パッチを示す。なお、 \mathbf{p} と \mathbf{q} のパッチサイズは $m \times m$ である。 $p(i, j)$ および $q(i, j)$ はパッチの i 行 j 列にある画素値を示す。

PatchMatch アルゴリズムはランダム探索アルゴリズムであるので、PatchMatch アルゴリズムを K 回繰り返すことにより、KNN 探索が実現できるように考えられる。しかしながら、注目パッチと同一画像から PatchMatch を行うと類似パッチとして、注目パッチ自身が探索されるという問題がある。また、注目パッチと異なる画像に対して PatchMatch を行った場合、同一のパッチが複数回類似パッチとして出力されるという問題点がある。

3.2. 空間的な距離を考慮した評価関数の提案

そこで本論文では、PatchMatch を行う際の評価関数として SSD だけでなく、既に探索された類似パッチとの空間的な距離を考慮した式(2)の評価関数を提案する。既に探索された類似パッチとの空間的な距離を考慮することにより、重複なく K 個の類似パッチの探索ができる。

$$E(\mathbf{p}, \mathbf{q}, \mathbf{r}) = \text{SSD}(\mathbf{p}, \mathbf{q}) + \omega(\mathbf{q}, \mathbf{r}) \quad \dots \text{式 (2)}$$

$E(\mathbf{p}, \mathbf{q}, \mathbf{r})$ は提案する評価関数であり、 \mathbf{r} は既に探索された類似パッチの集合である。

$\text{SSD}(\mathbf{p}, \mathbf{q})$ は式(1)を示しており、パッチ間の類似度を評価する項である。 $\omega(\mathbf{q}, \mathbf{r})$ は探索パッチと既に探索された類似パッチの空間的な距離を評価する項であり、式(3)で表わされる。

$$\omega(\mathbf{q}, \mathbf{r}) = \alpha \cdot \sum_{k=0}^K \exp\left(-\frac{D(\mathbf{q}, \mathbf{r}^k)}{2\sigma^2}\right) \quad \dots \text{式 (3)}$$

ここで、

$$D(\mathbf{q}, \mathbf{r}^k) = (q_x - r_x^k)^2 + (q_y - r_y^k)^2$$

であり、 q_x, q_y は探索パッチの画像座標である。 r_x^k, r_y^k はすでに発見した k 番目のパッチの画像座標である。

$D(\mathbf{q}, \mathbf{r}^k)$ は探索パッチ \mathbf{q} とすでに発見した k 番目のパッチとの空間的な距離を計算する関数である。式(3)中の、 α と σ はパラメータであり、本論文では、 α をパッチ内の画素数とし、 σ は 2.0 に設定している。

式(3)の評価関数を用いて PatchMatch による探索を K 回繰り返し行うことにより、重複のない K 個の類似パッチを探索することが可能となる。図1 (a) は注目パッチを青い四角形で囲む領域として示している。図1 (b) は提案手法により探索された K 個のパッチを赤い四角形で囲む領域で示している。なお、本論文では K は 16 である。図1

(c) は SSD 評価のみで処理した結果である。図より、提案手法により重複なく K 個のパッチが探索できていることがわかる。

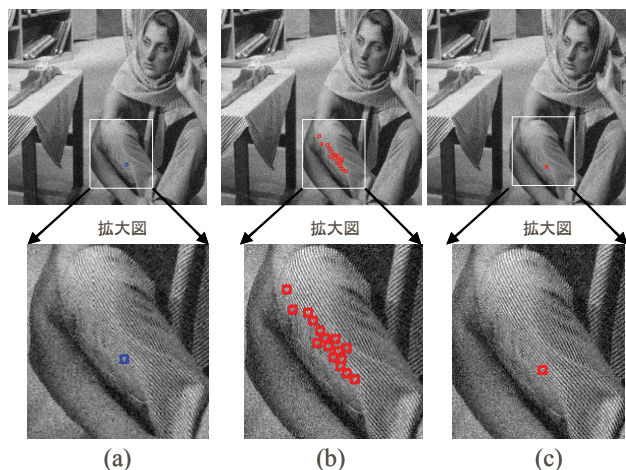


図1 注目パッチと探索された K 個のパッチ

4. Non-Local Means フィルタへの応用とその効果

本論文では、PatchMatch を用いた高速 KNN 探索を Non-Local Means フィルタへと応用した。提案手法を利用した場合と、従来の全探索を利用した場合について、処理時間と画質を比較し、提案手法の有効性を確認する。ところで、Non-Local Means フィルタは画像のパターンの類似度の高い部分の重みが大きいため、画像のエッジ部分は残しつつ、ノイズ除去ができる有効なフィルタである。

本論文では、注目パッチと K 個の輝度類似度を重みとして、画素値の加重平均を計算する。 K 個のパッチの加重平均によるノイズ除去の結果を Non-Local Means フィルタに適用し、全探索手法と本提案手法が画質へ与える影響について比較する。

本論文で提案した PatchMatch を用いた KNN 高速探索には反復処理があり、反復処理の回数が多ければ多いほど PatchMatch の結果は収束に向かう。本検討では、反復回数は 200 とする。表1に各手法の処理時間と PSNR の値を比較する。なお、各手法とも Intel@Core™2 Duo CPU P8700 2.53GHz の PC を使用した。表1より、 K 個のパッチの加重平均によるノイズ除去は、Non-Local Means より画質改善の効果があり、さらに、 K 個のパッチの探索に PatchMatch を用いることで処理時間が大幅に短縮されることがわかる。

表1 処理時間と PSNR の比較

	処理時間(min)	PSNR
Non-Local Means	1092	25.98
全探索による KNN 探索	786400	27.26
PatchMatch による KNN 探索	160	26.91

図2は PatchMatch の反復回数に対する PSNR の変化を示す図である。参考のために同図には全探索の結果を点線で併記している。PatchMatch は反復回数が増えるにつれ、画質が全探索の結果に近づいていく様子が確認できる。また、PatchMatch は初期の反復で急速に収束に向かうという性質があるため、近似類似 KNN 探索に適しているといえる。

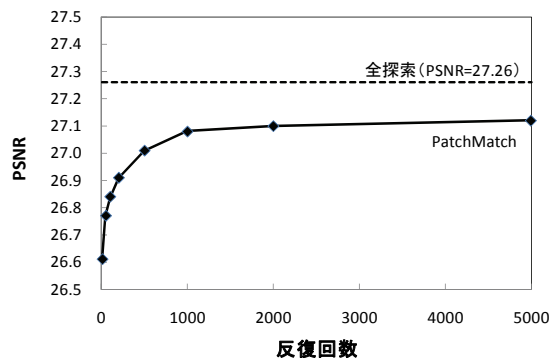


図2 PatchMatch の反復回数と PSNR

6. おわりに

本論文では、PatchMatch を用いた高速 KNN 探索方法のための、パッチ間の SSD と既に探索された類似パッチとの空間的な距離の両方を考慮した評価関数を提案した。これにより画像内の特定の箇所に集中することなく、重複のない K 個の類似パッチを高速に探索できることを示した。

参考文献

- [1] Connelly Barnes, et al., "PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing", SIGGRAPH2009, (2009).
- [2] A. Buades, et al., "A non-local algorithm for image denoising", IEEE Conference on Computer Vision and Pattern Recognition, vol.2, p.60, 2005.