

定義域分割法と Ridge 回帰を用いた 高精度 Rolling Guidance 画像フィルタ

深津 美薫^{1,2,a)} 吉澤 信^{2,b)} 竹村 裕^{1,c)} 横田 秀夫^{2,d)}

概要: デジタル画像中のテクスチャは様々な大きさで構成されており、顕著な特徴を保持しながら小さい構造を除去することは幅広い CGVI 応用に有用である。一方、既存法では畳み込み計算に由来するアーティファクトや収束性の問題が知られている。そこで本稿では、高品質なスケール対応画像フィルタ結果を生成する効率的な計算法を提案する。提案法は、Ridge 回帰の再帰適用で構成し、その畳み込み計算には、 L^1 ガウス関数を高速かつ高精度に畳み込む独自の定義域分割法を適用する。既存法と計算速度、近似精度、収束性に関して数値実験により比較を行い、約 7 桁~10 桁近似精度が高い高品質な結果を得た。

Accurate Rolling Guidance Image Filter via Domain Splitting and Ridge Regression Techniques

Abstract: Texture in a digital image consists of various scales, and removing its small structures while preserving salient image features is useful in a variety of CGVI applications. Conventional scale-aware image filters have some artifacts and convergence problems mainly caused by their inaccurate approximations of fast image convolutions. In this paper, we propose a novel, fast, and accurate computational framework to obtain high-quality scale-aware image filtering results. The framework is based on a recursive process of weighted ridge regressions and our domain-splitting technique which accurately approximates the L^1 Gaussian convolution quickly. We numerically examined our framework by comparing the existing filters in terms of speed, accuracy, and convergence rate, and achieved high-quality filtering results efficiently.

1. はじめに

デジタル画像の形状やテクスチャは、様々な大きさ（スケール: $\sigma \in \mathbb{R}_{>0}$ ）で構成されている。小さな構造は、テクスチャやノイズなどを表し、大きな構造は、境界線・空間的な色の変化・平坦な領域など大域的な構造を表している。そのため、画像解析にはスケールに応じて構造を分離することが重要であり、画像の領域抽出 [1]、物体検出 [2] や顕著度推定 [3] などの幅広い応用問題でスケールの分離が適用されている。信号・画像処理やコンピュータ・ビジョンの分野では、古くから、ガウシアンフィルタ（ガウス関数

とデータの線形畳み込み [4]）の標準偏差パラメータをスケールとして得られる画像群を考えるスケール空間理論 [5] が研究されている。また、古典的なスケール空間は線形に平滑化された画像群なので、画像の顕著なエッジ情報を同時に扱うことで、よりコンパクトに画像を表現できることが知られている [6]。一方、従来のバイラテラルフィルタ (BF: Bilateral Filter) [7] や非線形拡散フィルタ [8] などのエッジ保存平滑化フィルタ群は、指定したスケールに収束させることが容易ではない。特に、再帰的にそれらを適用した場合に、指定したスケールを大幅に超えて平滑化して定数に近づくか、そもそも収束しないため、繰り返し回数などのパラメータ調整と理解が難しい。

したがって、最近ではローリングガイダンスフィルタ (RG: Rolling Guidance) [9] に代表されるスケール対応フィルタ [9], [10], [11], [12], [13] が注目されている。スケール対応フィルタは与えられた目標のスケールより小さい構造を除去し、それより大きい顕著なエッジ（稜線構造）を復元し、目標のスケールに収束する。一般的なスケール対応

¹ 東京理科大学
Tokyo University of Science, 1-3 Kagurazaka, Shinjuku-ku,
Tokyo 162-8601, Japan

² 理化学研究所
RIKEN, 2-1, Hirosawa, Wako, Saitama 351-0198, Japan

a) miku.fukatsu@riken.jp

b) shin@riken.jp

c) takemura@rs.tus.ac.jp

d) hyokota@riken.jp

フィルタは、内部で結合フィルタと呼ばれる計算量が多い非線形畳み込みを繰り返し適用する必要があるため、実用的には高速計算法が必要である。既存のスケール対応フィルタでは、この高速計算法に Box 関数を畳み込み核に用いるガウシアンフィルタの近似法（移動平均法 [14] など）がよく用いられている。ガウシアンフィルタの高速近似法は高速 Fourier 変換 [15], 再帰フィルタ (Deriche 法 [16] など) や高速ガウス変換 [17], [18] など様々な方法が提案されており、その詳細は本稿の範囲を超えるため、文献 [19] を参照されたい。残念ながら Box 関数やガウス関数の単純打ち切りを用いた畳み込みは、Ringing アーティファクト [20] と呼ばれる偽のエッジを生成することが知られている [21]。これは、Box 関数やガウス関数の打ち切りは、周波数領域 (Fourier 変換) で $\frac{\sin(x)}{x}$ の波打った Sinc 関数と呼ばれる関数になり、指定したスケール (周波数) より大きな周波数が波打って周期的に強調されるためである (打ち切りのない完全なガウス関数ではこのような問題は起こらない [22]) (図 1, 図 2 参照*)。一方、統計学やその実験科学応用において、正規分布のデータ (すなわち L^2 ガウス関数) を近似するためには、半径 3σ を用いることが十分と考えられている。しかし、半径 3σ を単純に切り捨てたガウス関数の畳み込みでも、シャープなエッジ付近等で望ましくない偽のエッジが発生することが知られている [23]。

最近では、上記高速畳み込み計算法に起因するアーティファクトが無い高精度なスケール対応フィルタ [13] も提案されているが、内部で用いている定義域変換法 [24] の特性により、細長い領域などで収束性が悪い問題がある。また、近年の高階調・高解像画像・ディスプレイの普及、及びグラフィクス・画像処理技術の工学・自然科学への進展により、既存の計算法では達成が困難な近似精度が求められており、より高品質なスケール対応画像フィルタの開発が必要とされている。

そこで、本稿では高品質なフィルタ結果を効率的に生成するスケール対応画像フィルタの新しい計算法を提案し、先行研究との比較により、その性能と有効性を評価する。提案法は、スケール対応フィルタとして RG の結合フィルタを Ridge 回帰 (GF: Guided Filter [25] としても知られている) の再帰適用で構成する。初期平滑化と回帰計算内部の平滑化フィルタには、 L^1 ガウス関数を高速かつ超高精度に畳み込む定義域分割法 (DS: Domain Splitting) [23] を適応する (図 3 参照)。 L^1 ガウス関数による畳み込み (厳密には連続で滑らかな 1 次元の場合) は、線形スケール空間理論にて偽のエッジを生成しないことが知られている [22]。DS 法は、定義域を代表点群別に分割し、離散畳み込みを部分積分することで、従来要素数の 2 乗に比例す

*1 本稿での全ての勾配強度画像は輝度値の前進 1 次差分で近似した勾配の大きさを $\frac{1}{2}$ 乗して正規化表示している。

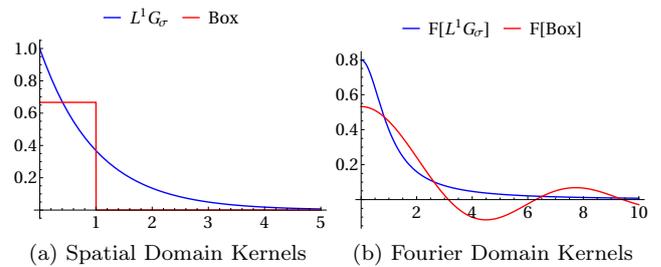


図 1: 畳み込み核: (a) は Box 関数 (赤) と L^1 ガウス関数 (青), (b) は対応する Fourier 変換: $F[\cdot]$.

Fig. 1 Box (red) and L^1 Gaussian (blue) kernels are plotted in the spatial (a) and its Fourier (b) domains.

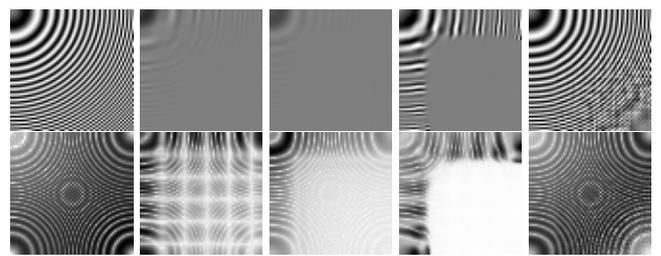


図 2: 既存法 RG-GF [9] の問題点と提案法 (Our-RGD) の比較。上段は単純平滑化 (b,c) とスケール対応フィルタ (d,e) の結果、下段は対応する勾配強度画像。Box 関数を用いる既存法 (d) は偽のエッジ (勾配強度の不連続線や歪み) が生成されるが L^1 ガウス関数を用いる提案法 (e) ではその様なアーティファクトは無い。

Fig. 2 Linear (b,c) and scale-aware (d,e) filtering results and their corresponding gradient magnitude (bottom).

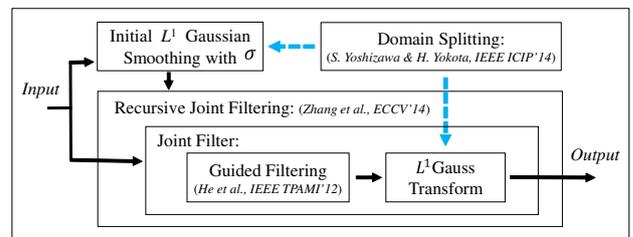


図 3: 提案法の概念図。

Fig. 3 Conceptual framework for our proposed filter.

る畳み込みの計算量を線形の計算量で高速近似する我々独自の手法である。代表点の間隔を理論的な数値安定条件から導くことで、高精度な近似が可能である。提案法を構成する回帰計算は等間隔画素の計算であり、非一様な画素を用いた先行研究 [13] と比べて、収束の一様性を高める。

以下本稿では、提案法の要素技術である RG, GF, 及び DS 法をそれぞれ第 2, 3 及び 4 章で記述する。第 5 章では提案法を導入し、第 6 章では数値実験による提案法の性能評価と既存法との比較を示す。第 7 章では本稿についてまとめ、提案法の制約と今後の展望について述べる。

Color-averaged MAE between adjacent iterations

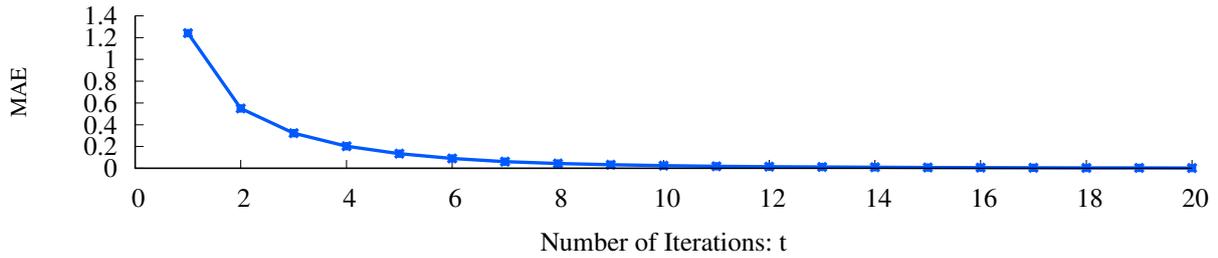


図 4: RG-BF の収束過程：繰り返し前後の平均絶対誤差（上段）とフィルタ結果例（下段）。ここで \mathbf{J}^t は t 回目の $\sigma = 5.0$ 及び JBF のエッジ制御パラメータ $\sigma_r = 1.5 \times \mathbf{I}$ を用いた結果。

Fig. 4 The top graph shows the color-averaged MAE (Mean Absolute Error) between adjacent iterations of RG-BF [9] with respect to the number of iterations (t), where the bottom images correspond to its filtering results.

2. ローリングガイダンス

Rolling Guidance (RG) フィルタ [9] は、詳細度を制御するスケール空間理論 [5] に基づいたスケール対応画像フィルタである。また、RG フィルタは、結合 BF (JBF: Joint BF) [26], [27] や GF [25] 等の入力画像へガイド画像を反映して平滑化する結合フィルタと呼ばれる方法を用いて、顕著なエッジ情報を復元しながら、ある特定のスケール σ よりも小さい構造を除去する。さらに、RG フィルタは指定したスケール構造への非常に高い収束性が知られており (図 4 参照)、特徴・テクスチャ抽出やスタイル化など様々なグラフィクス・画像処理応用に有用である。

RG フィルタは、まず初めに指定したスケール σ より小さな構造をガウシアンフィルタ式 (1) によって除去する (線形ガウシアンスケール空間 [5])。

$$\mathbf{J}^0(\mathbf{x}) \equiv \frac{1}{K(\mathbf{x})} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} \exp\left(-\frac{|\mathbf{x} - \mathbf{y}|^2}{2\sigma^2}\right) \mathbf{I}(\mathbf{y}). \quad (1)$$

ここで、画素座標は $\mathbf{x} \in \mathbb{R}^2$ 、入力画像のカラーは $\mathbf{I} = \mathbf{I}(\mathbf{x}) \in \mathbb{R}^3$ 、再帰計算の初期カラーは $\mathbf{J}^0 = \mathbf{J}^0(\mathbf{x}) \in \mathbb{R}^3$ 、 $K(\mathbf{x})$ は

$$K(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} \exp\left(-\frac{|\mathbf{x} - \mathbf{y}|^2}{2\sigma^2}\right)$$

で定義された正規化項、 $\exp(\cdot)$ は指数関数、 $\mathcal{N}(\mathbf{x})$ は画素 \mathbf{x} の近傍画素集合である。 σ よりも小さな構造は、このガウシアンフィルタによって取り除かれる。 σ より大きな構造は、平滑化されるが、完全に除去されることはない。

次に結合フィルタを繰り返し再帰的に適用し、顕著なエッジを復元する。 $\mathbf{J}^t = \mathbf{J}^t(\mathbf{x}) \in \mathbb{R}^3$ を RG によって更新される t 番目のカラーとすると、 $(t + 1)$ 回適用のカラー \mathbf{J}^{t+1} は、入力 \mathbf{I} はそのまま、結合フィルタ内の \mathbf{J}^t を再帰的にガイド画像として更新する。 図 4 に JBF を用いた場合 (RG-BF) の \mathbf{J}^t と \mathbf{J}^{t+1} の平均絶対誤差 (MAE: Mean Absolute Error) [9] と対応するフィルタ結果例を示す。 繰り返し回数を増やすと MAE が零に近づき、出力画像が収束していることが分かる。 また、単純平滑化の \mathbf{J}^0 から繰り返し回数 t が進むにつれて ($t = 4$ でも実用的とされている [9]) 顕著なエッジが復元されていることが確認できる。 一方、JBF はデータ依存の非線形畳み込みであるため、高精度に高速計算を行うことは容易ではない [28]。 そこで、提案法では、上記 JBF の代わりに次章で紹介する線形 Ridge 回帰を結合フィルタに用いる。

3. ガイド付きフィルタ (Ridge 回帰)

ガイド付きフィルタ (GF: Guided Filter) [25] は、入力画像とは別のガイド画像と呼ばれる参照用画像の影響を線形 Ridge 回帰 [29], [30] を用いて入力画像へ反映するフィルタである。 また、入力画像とガイド画像を同じにすることで、BF のようにエッジを保持した平滑化特性を持つ。 GF は局所的な線形変換を考えているため、BF と比べて勾配反転と呼ばれるアーティファクトが発生しにくい [25]。 さらに、線形平滑化フィルタの組み合わせで構成できるため、BF 等と比べて、移動平均法 [14] などを用いて比較的容易に高速計算法を適用することが可能である。

画像座標 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$ に対応する入力画像の輝度値を $I = I(\mathbf{x}) \in \mathbb{R}$ とすると, GF は, その出力画像の輝度値 $q = q(\mathbf{x}) \in \mathbb{R}$ をガイド画像の輝度値 $h = h(\mathbf{x}) \in \mathbb{R}$ の線形変換であると仮定して, 式 (2) を考える.

$$q(\mathbf{x}) = a(\mathbf{y})h(\mathbf{x}) + b(\mathbf{y}), \quad \forall \mathbf{x} \in \mathcal{N}(\mathbf{y}). \quad (2)$$

ここで, $\mathcal{N}(\mathbf{y})$ は画素 \mathbf{y} の近傍画素集合であり, $a = a(\mathbf{y}) \in \mathbb{R}$, $b = b(\mathbf{y}) \in \mathbb{R}$ は $\mathcal{N}(\mathbf{y})$ 内で一定な係数で, 入力画像 I とガイド画像 h から求まる. また, 式 (2) の両辺に勾配作用素を適用すると, $\mathcal{N}(\mathbf{y})$ 内では $\nabla q = a\nabla h$ となる. したがって, エッジ (画像内の稜線構造) は平行移動せず, 勾配反転アーティファクトが軽減される [25]. 係数 (a, b) は以下の式 (3) を最小化し求める. $E(a(\mathbf{y}), b(\mathbf{y})) =$

$$\sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} (a(\mathbf{y})h(\mathbf{x}) + b(\mathbf{y}) - I(\mathbf{x}))^2 + \epsilon a(\mathbf{y})^2. \quad (3)$$

ここで, ϵ は a が大きくなりすぎないようにする正規化項の制御パラメータである. 式 (3) は線形 Ridge 回帰 [29], [30] として知られており, 以下のように (a, b) が求まる.

$$a(\mathbf{y}) = \frac{\frac{1}{|\mathcal{N}|} \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} h(\mathbf{x})I(\mathbf{x}) - \bar{h}(\mathbf{y})\bar{I}(\mathbf{y})}{\epsilon + V^2(\mathbf{y})},$$

$$b(\mathbf{y}) = \bar{I}(\mathbf{y}) - a(\mathbf{y})\bar{h}(\mathbf{y}).$$

ここで, $|\mathcal{N}|$ は $\mathcal{N}(\mathbf{y})$ 内の画素数, $\bar{h}(\mathbf{y})$ と $V^2(\mathbf{y})$ は $\mathcal{N}(\mathbf{y})$ における $h(\mathbf{x})$ の平均 $\bar{h}(\mathbf{y}) = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} h(\mathbf{x})$ と分散 $V^2(\mathbf{y}) = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} h(\mathbf{x})^2 - \frac{1}{|\mathcal{N}|^2} (\sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} h(\mathbf{x}))^2$, $\bar{I}(\mathbf{y})$ は入力画像 $I(\mathbf{x})$ の平均 $\bar{I}(\mathbf{y}) = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{y})} I(\mathbf{x})$ である.

係数 (a, b) はいくつかの異なる矩形領域に重複して含まれていて, これらの値は一意ではないので, 文献 [25] では近傍の平均を Box 関数 (移動平均法 [14]) で推定している. 最終的な出力画像の輝度値は, 式 (4) となる.

$$q(\mathbf{x}) = \bar{a}(\mathbf{x})h(\mathbf{x}) + \bar{b}(\mathbf{x}). \quad (4)$$

このとき, $\bar{a}(\mathbf{x}) = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} a(\mathbf{y})$ 及び $\bar{b}(\mathbf{x}) = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} b(\mathbf{y})$ である.

図 5 には入力・ガイド画像共にカラー画像を用いた場合の単純再帰適用例 (RG ではない) を示す. 出力画像はガイド画像が反映されていることが分かる.

4. 定義域分割法

定義域分割法 (DS: Domain Splitting) [23] はガウス関数の畳み込みを正確かつ高速に近似する. DS 法は, L^2 ノルムの代わりに L^1 ノルムを用いて, 畳み込みの積分領域を領域上の代表点を基準とした部分領域に分割し, その部分積分を効率的かつ高精度に近似する. 本稿で提案する計算法では, ガウシアンフィルタ (式 (1) の \mathbf{J}^0) と結合フィルタで用いる GF 内の平滑化フィルタ $f(\cdot)$ にこの定義域分割法を適応する.

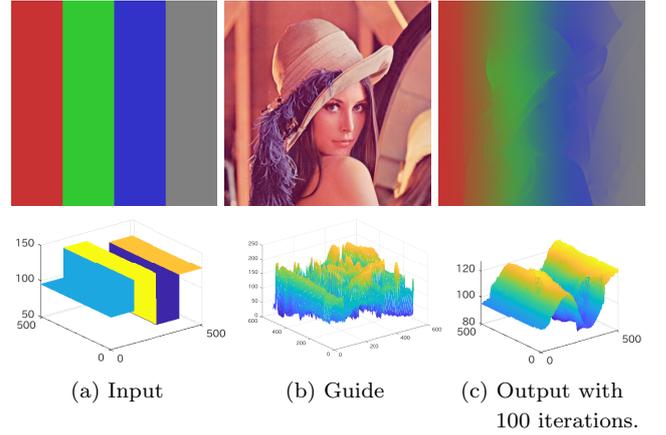


図 5: GF の単純再帰適用例 ($\sigma = 5.0$, $\epsilon = 0.01$).
Fig. 5 A recursive GF result (c) and its input (a) and guide (b) images, where $\sigma = 5.0$ and $\epsilon = 0.01$.

1 次元の画素を表す \mathbb{R} 上の n 個の点群 $\mathcal{P} : \{t_i\} (t_1 \leq t_2 \leq \dots \leq t_n)$ を考えると, $t_j \in \mathcal{P}$ における $h(\cdot)$ のガウス変換 $f(\cdot) \in \mathbb{R}$ は式 (5) で与えられる.

$$f(t_j) = \sum_{i=1}^n G_\sigma(t_j - t_i)h(t_i). \quad (5)$$

ここで, すべての $j = \{1, 2, \dots, n\}$ について, 単純に $f(t_j)$ を計算すると, 計算量は n の二乗に比例する.

4.1 定義域分割法の基本概念

固定点 t_1 に対して, t_i と t_j の L^1 ノルム距離をそれらの順序を考慮して分割すると,

$$|t_i - t_j| = \begin{cases} |t_i - t_1| - |t_j - t_1| & \text{if } t_1 \leq t_j \leq t_i, \\ -|t_i - t_1| + |t_j - t_1| & \text{if } t_1 \leq t_i < t_j. \end{cases}$$

となる. 上式を L^1 ガウス関数 $G_\sigma(t_i - t_j)$ に代入すると, 指数関数の性質より,

$$G_\sigma(t_i - t_j) = \begin{cases} \frac{G_\sigma(t_i - t_1)}{G_\sigma(t_j - t_1)} & \text{if } t_1 \leq t_j \leq t_i, \\ \frac{G_\sigma(t_j - t_1)}{G_\sigma(t_i - t_1)} & \text{if } t_1 \leq t_i < t_j \end{cases}$$

となる. ここで, ガウス関数内の 2 つの添え字 i と j の依存関係は, 2 変数関数が 2 つの 1 変数関数の組み合わせで置き換えられていることで, 解消されていることに注目されたい. このとき, L^1 ガウス変換 $f(\cdot)$ は結果として, 式 (6) となる.

$$f(t_j) = h(t_j) + G_\sigma(t_j - t_1)\xi(j-1) + \frac{\eta(j+1)}{G_\sigma(t_j - t_1)},$$

$$\xi(j) = \sum_{i=1}^j \frac{h(t_i)}{G_\sigma(t_i - t_1)}, \quad \eta(j) = \sum_{i=j}^n G_\sigma(t_i - t_1)h(t_i). \quad (6)$$

ここで, $\xi(0) \equiv 0 \equiv \eta(n+1)$ である. なお, 式 (6) は添え字 j のみに依存し, $\xi(\cdot)$ と $\eta(\cdot)$ は $f(t_j)$ を計算する前にす

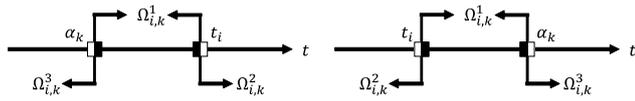


図 6: $\Omega_{i,k}^1, \Omega_{i,k}^2, \Omega_{i,k}^3$ の領域.

Fig. 6 Division of the domains $\Omega_{i,k}^1, \Omega_{i,k}^2, \Omega_{i,k}^3$ via the poles $\{\alpha_k\}$.

すべての $j = \{1, 2, \dots, n\}$ について, 事前に 1 回計算するだけでよい (線形計算複雑度). したがって, 式 (6) の分解によって, すべての添え字 j について $f(t_j)$ を計算するためには, 線形の計算複雑度 $O(3n)$ となる. しかし, この計算は数値的に不安定なことがあり, 式 (6) を正確に計算することは容易ではない. 例えば, $\frac{1}{G_\sigma(t_i - t_1)}$ は, $|t_i - t_1|$ が非常に大きな値の場合, オーバーフローを起こす.

4.2 代表点群を用いた高精度近似

上記の数値問題を回避するため, 固定点 t_1 を用いる代わりに \mathbb{R} 上の m 個の代表点群 $\{\alpha_k\}$ を考える. ここで, $\alpha_1 < \alpha_2 < \dots < \alpha_m$ と仮定すると, α_k の周りの $|t_i - t_j|$ の定義域分割は

$$|t_i - t_j| = \begin{cases} |t_i - \alpha_k| - |t_j - \alpha_k| & \text{if } t_j \in \Omega_{i,k}^1, \\ -|t_i - \alpha_k| + |t_j - \alpha_k| & \text{if } t_j \in \Omega_{i,k}^2, \\ |t_i - \alpha_k| + |t_j - \alpha_k| & \text{if } t_j \in \Omega_{i,k}^3. \end{cases}$$

となる. ここで, 領域 $\Omega_{i,k}^1, \Omega_{i,k}^2, \Omega_{i,k}^3$ は図 6 のように

$$\begin{aligned} \Omega_{i,k}^1 &= \{z \in \mathcal{P} : \alpha_k \leq z \leq t_i, \text{ or } t_i \leq z \leq \alpha_k\}, \\ \Omega_{i,k}^2 &= \{z \in \mathcal{P} : \alpha_k \leq t_i < z, \text{ or } z < t_i \leq \alpha_k\}, \\ \Omega_{i,k}^3 &= \{z \in \mathcal{P} : z < \alpha_k \leq t_i, \text{ or } t_i \leq \alpha_k < z\} \end{aligned}$$

と定義される. この代表点群 $\{\alpha_k\}$ の定義域分割により, 式 (5) のガウス変換 $f(\cdot)$ は

$$f(t_j) = h(t_j) + C_j + D_j + E_j, \quad (7)$$

$$C_j = \left\{ G_{(\sigma,j,\gamma(j))} \sum_{i=\gamma_2(\gamma(j))}^{j-1} \frac{h(t_i)}{G_{(\sigma,i,\gamma(j))}} \right\} + \left\{ \frac{1}{G_{(\sigma,j,\gamma(j))}} \sum_{i=j+1}^{\gamma_2(\gamma(j)+1)-1} G_{(\sigma,i,\gamma(j))} h(t_i) \right\}, \quad (8)$$

$$D_j = \sum_{k=1}^{\gamma(j)-1} G_{(\sigma,j,k)} A_k, \quad E_j = \sum_{k=\gamma(j)+1}^m G_{(\sigma,j,k)} B_k,$$

$$A_k = \sum_{i=\gamma_2(k)}^{\gamma(k+1)-1} \frac{h(t_i)}{G_{(\sigma,i,k)}}, \quad B_k = \sum_{i=\gamma_2(k)}^{\gamma(k+1)-1} G_{(\sigma,i,k)} h(t_i) \quad (9)$$

となる. ここで, $G_{(\sigma,l,k)} \equiv G_\sigma(t_l - \alpha_k)$, $\alpha_k \leq t_j < \alpha_{k+1}$ となるような $\gamma(j) = k$, $\gamma_2(k) = \min(j)$ である (図 7 参照).

計算精度の上限 δ が与えられた場合, 前述した数値不安定性を避けるために, 不等式 $\exp\left(\frac{|\alpha_{k+1} - \alpha_k|}{\sigma}\right) < \delta$ を満たさ

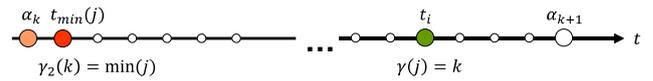


図 7: インデックス関数の概念図.

Fig. 7 Indexing functions $\gamma(\cdot)$ and $\gamma_2(\cdot)$.

なければならない. この条件を満たすことで, $\varphi \in (0, 1)$ をパラメータとした関係式 $|\alpha_{k+1} - \alpha_k| = \varphi \sigma \log(\delta)$ が導かれ, 隣り合う代表点間の距離が数値安定条件を満たす様に求められる. 本稿では文献 [23] に従い, $\varphi = 0.5$ とし, δ は, 倍精度浮動小数点の上限を使用した (64-bit C++ プログラミング言語の DBL_MAX). \mathcal{P} の範囲は, $w = t_n - t_1 > 0$ と定義されるので, 代表点の数とその座標は自動的に式 (10) で与えられる.

$$\{\alpha_k\} = t_1 + \{0, 1, \dots, m-1\} \frac{w}{m}, \quad m = \left\lceil \frac{w}{\varphi \sigma \log(\delta)} \right\rceil. \quad (10)$$

ここで, $[\cdot]$ は天井関数であり, 画像処理の一般的な場合に $m \ll n$ である. 本稿では, t_1 は常に 0 であるが, 式 (10) のように, 1 つ目の代表点 ($\alpha_1 = t_1$) を移動させると, 代表点群 $\{\alpha_k\}$ は安定条件を満たす様に選んでいるため, 式 (7) が数値的に安定することが数学的に保障される.

さらに, $|\alpha_k - t_j| > \sigma \log(\delta)$ となると, α_k が t_j から遠いところでは, $G_\sigma(\alpha_k - t_j)$ は数値的に零になる. したがって, D_j と E_j は式 (9) の代わりに,

$$\begin{aligned} D_j &\approx G_{(\sigma,j,\gamma(j)-1)} A_{\gamma(j)-1}, \\ E_j &\approx G_{(\sigma,j,\gamma(j)+1)} B_{\gamma(j)+1} \end{aligned} \quad (11)$$

と高精度に近似できる. ここで, $\gamma(j) < 2$ のとき, $D_j \approx 0$ であり, $\gamma(j) > n-1$ では, $E_j \approx 0$ である. t_j での近似誤差は解析的に求められ,

$$\sum_{k=1}^{\gamma(j)-2} G_\sigma(t_j - \alpha_k) A_k + \sum_{k=\gamma(j)+2}^m G_\sigma(t_j - \alpha_k) B_k$$

となる. したがって, 上記の $f(t_j)$ の近似は, $3\varphi \sigma \log(\delta)$ の領域内で和を切り捨てるのと同等の精度である. このような切り捨ては非常に正確で, $\varphi = 0.5$, $\log(\text{DBL_MAX}) \approx 700$ の半径が約 525σ の単純な打ち切りと同等である (実験科学分野ではよく 3σ の近似が用いられている). また, この定義域分割法では, $O(4n + 2m + \max(n, m))$ の計算複雑度 (σ に関して, $O(1/[\sigma])$) しか必要ない. 上記 1 次元の定義域分割法は, **分離実装** で 2 次元画像に適用される. 分離実装は, 画像を画素の行列としたとき, 行だけ 1 次元の畳み込みをして, その結果に列だけの 1 次元畳み込みを行う方法であり, 畳み込み核がガウス関数の様に次元別に分離可能である場合は, 2 次元畳み込みと同義な結果を与える. 本研究では, 文献 [13] のアルゴリズムにしたがって, 式 (7)-式 (11) を実装した.

5. 提案法: Our-RGD フィルタ

本章では提案法 Our-RGD (RG via GF with DS) を記述する. 先行研究の GF [25] や RG-GF [9] では, 本稿第3章で説明したとおり, 式 (4) のように係数を $(\bar{a}(\mathbf{x}), \bar{b}(\mathbf{x}))$ と Box 関数で平滑化する必要がある. 一方 DS 法を用いた場合, 係数の平滑化が GF では不要なことが知られている [31]. そこで, 提案法では結合フィルタに係数を平滑化しない DS 法に基づく GF (GF-DS) を用いる. したがって, Our-RGD を $(t+1)$ 回適用したカラー値 $\mathbf{J}^t = \{J_v^t(\mathbf{x})\} \in \mathbb{R}^3$, $v \in \{r, g, b\}$ は式 (12) で与えられる.

$$J_v^{t+1} = \mathbf{a}_v^T \mathbf{J}^t + b_v \quad v \in \{r, g, b\}. \quad (12)$$

ここで $\mathbf{a} = \mathbf{a}(\mathbf{y}) \in \mathbb{R}^3 \times \mathbb{R}^3$ は, 入力 $\mathbf{I} = \mathbf{I}(\mathbf{x}) \in \mathbb{R}^3$ とガイド画像 \mathbf{J}^t のカラーチャンネル数を掛けた個数の係数

$$\mathbf{a} = (\mathbf{a}_r, \mathbf{a}_g, \mathbf{a}_b) = \begin{pmatrix} a_{r,r} & a_{g,r} & a_{b,r} \\ a_{r,g} & a_{g,g} & a_{b,g} \\ a_{r,b} & a_{g,b} & a_{b,b} \end{pmatrix}, \quad (13)$$

$\mathbf{b} = \mathbf{b}(\mathbf{y}) \in \mathbb{R}^3$ の要素は $\mathbf{b} = (b_r, b_g, b_b)$ である. このとき, 線形係数 (\mathbf{a}_v, b_v) は, L^1 ガウス関数 $G_\sigma(\cdot) = \exp(-\frac{|\cdot|}{\sigma})$ を重みに用いて, \mathbf{I} の要素を $\mathbf{I} = (I_r, I_g, I_b)$ とした各要素毎 ($v \in \{r, g, b\}$) の目的関数

$$E(\mathbf{a}_v(\mathbf{y}), b_v(\mathbf{y})) = \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{x})} G_\sigma(|\mathbf{x} - \mathbf{y}|) ((\mathbf{a}_v^T(\mathbf{y}) \mathbf{J}^t(\mathbf{x}) + b_v(\mathbf{y}) - I_v(\mathbf{x}))^2 + \epsilon |\mathbf{a}_v(\mathbf{y})|^2),$$

を最小化することで, 式 (14), (15) で与えられる.

$$\mathbf{a}_v = (\mathbf{C} + \epsilon \mathbf{U})^{-1} (f(I_v) \mathbf{J}_v^t - f(I_v) f(\mathbf{J}_v^t)), \quad (14)$$

$$b_v = f(I_v) - \mathbf{a}_v^T f(\mathbf{J}^t). \quad (15)$$

ここで, f は被積分関数 $\mathbf{h} = \mathbf{h}(\mathbf{x})$ と L^1 ガウス関数の DS 法による正規化畳み込み

$$f(\mathbf{h}) \equiv \frac{1}{\sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} G_\sigma(|\mathbf{x} - \mathbf{y}|)} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} G_\sigma(|\mathbf{x} - \mathbf{y}|) \mathbf{h}(\mathbf{y}), \quad (16)$$

\mathbf{U} は 3×3 の単位行列, \mathbf{C} は, \mathbf{J}^t の期待値をフィルタ f で見積もった共分散行列 $\mathbf{C} =$

$$\begin{pmatrix} f(J_r^t) - (f(J_r^t))^2 & f(J_r^t J_g^t) - f(J_r^t) f(J_g^t) & f(J_r^t J_b^t) - f(J_r^t) f(J_b^t) \\ f(J_g^t J_r^t) - f(J_g^t) f(J_r^t) & f(J_g^t) - (f(J_g^t))^2 & f(J_g^t J_b^t) - f(J_g^t) f(J_b^t) \\ f(J_b^t J_r^t) - f(J_b^t) f(J_r^t) & f(J_b^t J_g^t) - f(J_b^t) f(J_g^t) & f(J_b^t) - (f(J_b^t))^2 \end{pmatrix},$$

ϵ は正規化項の制御パラメータであり, 実用上は ϵ に輝度値の値域 (最大と最小の差) \mathcal{L} の 2 乗をかけた値 ($\epsilon \rightarrow \epsilon \mathcal{L}^2$) を用いる.

RG 内の結合フィルタとして再帰的に用いる場合に, 入力画像のカラー値 \mathbf{I} 及びその平滑化結果 $f(\mathbf{I})$ は Our-RG の初期状態 $\mathbf{J}^0 \equiv f(\mathbf{I})$ で用いた値を再利用出来る. この再利用と DS 法を組み合わせて効率的な工夫をした重み

Algorithm 1: GF-DS

Input : Pixels $\{\mathbf{x}\}$ and their corresponding input image colors $\mathbf{I} = \{I_v(\mathbf{x}), v \in \{r, g, b\}\}$, guide image colors $\mathbf{h} = \{h_v(\mathbf{x}), v \in \{r, g, b\}\}$, smoothed input colors $\mathbf{mean}_\mathbf{I} = \{f(I_v), v \in \{r, g, b\}\}$, regularization ϵ , and scale σ , where s_x and s_y are the image width and height, respectively.

Output: Filtered colors $\{\mathbf{q} = \{q_v(\mathbf{x}), v \in \{r, g, b\}\}$.

- 1 **Function** GF-DS($\mathbf{I}, \mathbf{h}, \mathbf{mean}_\mathbf{I}, \epsilon$):
- 2 **forall** $v \in \{r, g, b\}$ **do**
- 3 $mean_{h_v} \leftarrow f(h_v)$; // DSSum $\times (s_x + s_y)$.
- 4 $corr_{h_v} \leftarrow f(h_v \times h_v)$; // DSSum $\times (s_x + s_y)$.
- 5 $corr_{h, I_v} \leftarrow \{f(h_r \times I_v), f(h_g \times I_v), f(h_b \times I_v)\}$; // DSSum $\times 3 \times (s_x + s_y)$.
- 6 **end forall**
- 7 $mean_{h_r, h_g} \leftarrow f(h_r \times h_g)$; // DSSum $\times (s_x + s_y)$.
- 8 $mean_{h_r, h_b} \leftarrow f(h_r \times h_b)$; // DSSum $\times (s_x + s_y)$.
- 9 $mean_{h_g, h_b} \leftarrow f(h_g \times h_b)$; // DSSum $\times (s_x + s_y)$.
- 10 **forall** \mathbf{x} **do**
- 11 // The 3×3 covariance matrix.
- 12 $A_{11} \leftarrow corr_{h_r} - mean_{h_r} \times mean_{h_r}$;
- 13 $A_{12} \leftarrow mean_{h_r, h_g} - mean_{h_r} \times mean_{h_g}$;
- 14 $A_{13} \leftarrow mean_{h_r, h_b} - mean_{h_r} \times mean_{h_b}$;
- 15 $A_{22} \leftarrow corr_{h_g} - mean_{h_g} \times mean_{h_g}$;
- 16 $A_{23} \leftarrow mean_{h_g, h_b} - mean_{h_g} \times mean_{h_b}$;
- 17 $A_{33} \leftarrow corr_{h_b} - mean_{h_b} \times mean_{h_b}$;
- 18 $var_{\mathbf{h}} \leftarrow \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{12} & A_{22} & A_{23} \\ A_{13} & A_{23} & A_{33} \end{pmatrix}$;
- 19 **forall** $v \in \{r, g, b\}$ **do**
- 20 $cov_{h I_v} \leftarrow corr_{h, I_v} - mean_{I_v} \times mean_{\mathbf{h}}$;
- 21 $\mathbf{a}_v \leftarrow (var_{\mathbf{h}} + \epsilon \mathbf{U})^{-1} cov_{h I_v}$;
- 22 $b_v \leftarrow mean_{I_v} - \mathbf{a}_v \times mean_{h_v}$;
- 23 $q_v \leftarrow \mathbf{a}_v \times \mathbf{h} + b_v$;
- 24 **end forall**
- 25 **end forall**
- 26 **return** $\{\mathbf{q}\}$;

Algorithm 2: Our-RGD

Input : Pixels $\{\mathbf{x}\}$ and their corresponding input image colors $\mathbf{I} = \{I_c(\mathbf{x}), c \in \{r, g, b\}\}$, regularization ϵ , scale σ , image range $\mathcal{L} = |\max(\mathbf{I}) - \min(\mathbf{I})|$, and iteration number t .

Output: Filtered colors $\{\mathbf{J}^t = \{J_c^t(\mathbf{x}), c \in \{r, g, b\}\}$.

- 1 Initialize $f(\cdot)$; // (DSInit, DSSum) $\times 2$.
- 2 $\epsilon \leftarrow \epsilon \times \mathcal{L}^2$;
- 3 $\mathbf{J}^0 \leftarrow \mathbf{mean}_\mathbf{I} \leftarrow f(\mathbf{I})$; // DSSum $\times 3 \times (s_x + s_y)$.
- 4 **for** $i \leftarrow 1$ **to** t **do**
- 5 $\mathbf{J}^{i+1} \leftarrow$ GF-DS($\mathbf{I}, \mathbf{J}^i, \mathbf{mean}_\mathbf{I}, \epsilon$); // Algorithm 1.
- 6 **end for**
- 7 **return** $\{\mathbf{J}^t\}$;

表 1: 既存法 RG-GF [9] と提案法の内部で用いている平滑化フィルタ回数の比較.

Table 1 Comparison of the number of f employed in RG-GF [9] and Our-RGD.

	RG-GF [9]	Our-RGD
The number of f	$3+33t$	$3+18t$

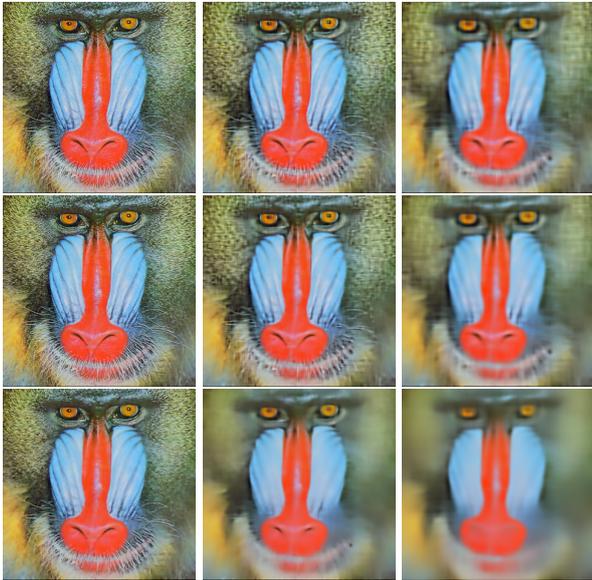


図 8: 様々なパラメータによる Our-RGD の結果 ($\sigma \in \{2, 8, 16\}$ (横), $\epsilon \in \{0.01, 0.001, 0.0001\}$ (縦), $t = 20$).

Fig. 8 Various scale and regularization parameters of Our-RGD results, where $\sigma \in \{2, 8, 16\}$ (row), $\epsilon \in \{0.01, 0.001, 0.0001\}$ (column), and $t = 20$.

付き GF-DS の疑似コードを Algorithm 1 に示す. 既存の RG-GF [9] と比べて提案法は内部で用いる $f(\cdot)$ の数が少ない利点がある (表 1 参照). ここで, Algorithm 1 中の $mean$ は f による期待値, $corr$ は相関, var は分散, cov は共分散の略である. また, Our-RGD の疑似コードは Algorithm 2 であり, パラメータ変化をさせたときのフィルタ結果例を図 8 に示す.

6. 数値実験による性能評価

本章では, 数値実験により速度, 近似精度, フィルタ品質, 及び収束度合に関して提案法と既存法を比較した結果を示す. 表 2 のように, 4 つの既存法との比較を実施した. ここで, Box 関数を用いて結合フィルタに GF, WGF [32] を用いたフィルタをそれぞれ RG-GF [9], RG-WGF*2 とし, 結合フィルタに JBF を用いて高速計算法には Permutohedral Lattice [33] を用いる方法 (文献 [9] の著者が公開しているコードを使用) を RG-BF, DS 法を用いて結合フィルタに定義域変換法 (DT: Domain Transform) [24] を用いたフィ

*2 効率化のため, RG-GF 及び RG-WGF は提案法と同様に $f(\mathbf{I})$ を再利用する実装を本稿第 6.1 節の実行時間計測も含めて用いた (表 1 の RG-GF 内の f が $3 + 30t$ 回の実装).

表 2: 既存法と提案法.

Table 2 Conventional and our filters.

	Conventional				Our
	RG-GF	RG-WGF	RG-BF	RG-DT-DS	Our-RGD
$f(\cdot)$	Box (Moving Average)	Permutohedral L.	DS		
Joint filter	GF	WGF	BF	DT	GF
Ref.	[9]	[32]	[9]	[13]	This paper

				
	Step-edge	Mandrill	Lena	Snack
w	128	512	512	1368
h	128	512	512	912

図 9: 本稿第 6 章で使用した入力画像. w と h はそれぞれ縦と横のピクセル数である.

Fig. 9 Input color images used in Sec. 6 of this paper where w (width) and h (height) stand for the numbers of row and column pixels, respectively.

表 3: フィルタが 1 秒間に処理できる画素数 (megapixels: 10^6) の比較. t は繰り返し回数.

Table 3 Speed comparison of our and conventional filters (megapixels per second), where t represents the iteration number.

RG-GF	RG-WGF	RG-BF	RG-DT-DS	Our-RGD
$1.53/t$	$1.50/t$	$1.76/t$	$2.29/t$	$1.24/t$

ルタを RG-DT-DS [13] とした. 数値実験には, ランダムに生成した色を持つ画像群と図 9 に示す 4 つの入力画像を用いた. 以降の数値実験では, Box 関数を畳み込み核とした高速畳み込みは, 移動平均法 [14] を用いてその半径 r は, 文献 [19], [34] に従って式 (17) とした.

$$r = \frac{\sqrt{12\sigma^2 + 1}}{2} - 1 \quad (17)$$

また, 全ての実験で 64-bit C++ プログラミング言語にて実装を行い, コンパイラには GNU g++9.3.0, AMD Ryzen 7 3700X 8-Core Processor の CPU, メモリ 32GB の PC を使用した.

6.1 計算速度比較

サイズが異なる画像に対して, それぞれのフィルタを本稿第 6.2 節でも使用する 24 通りのパラメータセット ($\sigma \in \{4, 8, 16, 32\}$, $\epsilon \in \{0.5, 0.01, 0.05, 0.001, 0.005, 0.0001\}$) に対して処理時間を計測した. 表 3 はそれぞれのフィルタが 1 秒間に処理できる画素数であり, サイズと繰り返し回数を横軸にとった平均計算時間 (秒) のグラフを図 10 に示す. いずれのフィルタでも画素数と実行時間は線形の関係であることが分かる. 提案法は既存法よりも少し遅いが既存法と同等の実用的な計算実行速度であることが確認できる. 参考までに f の実装を高速化無しのナイーブな

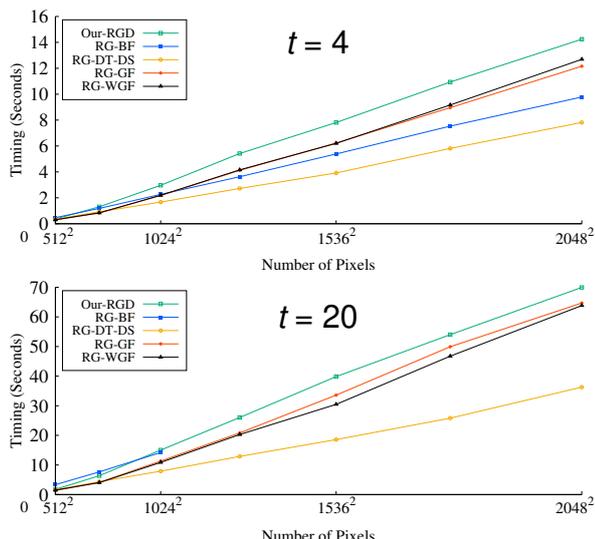


図 10: フィルタの平均計算時間 (縦軸: 秒, 横軸: 画素数, 上段: $t = 4$, 下段: $t = 20$).

Fig. 10 Timings (seconds) of our and conventional filters with respect to image size, where top and bottom graphs show the cases $t \in \{4, 20\}$, respectively.

畳み込みで行う RG-GF は, 512×512 画素を処理するのに ($t = 20$ のとき) 約 13.8 時間必要である.

6.2 近似精度比較

画像処理, コンピュータ・ビジョン, コンピュータ・グラフィックスで幅広く用いられている Box 関数 (移動平均法 [14]) 及び Deriche 法 [16] (再帰フィルタに基づく近似法) を \mathbf{J}^0 と結合フィルタ内の f : 式 (16) に用いた場合と提案法を比較し, 提案法の近似精度を定量評価した. Box 及び Deriche は L^2 ガウス関数の FIR (Finite Impulse Response) [19], DS 法は L^1 ガウス関数の FIR の結果をそれぞれ 64-bit 倍精度浮動小数点の精度で比較した. FIR 及び Deriche 法は公開ライブラリー [19] を用いて精度 $\text{tol} = 10^{-15}$ (及び Deriche は 4-th 再帰オーダー: $K = 4$) とした. 入力として, カラー値範囲を $[0, 1]$ としてランダムに生成した 64×64 画素サイズのカラー画像を 100 枚用いて, $\sigma \in \{4, 8, 16, 32\}$, $\epsilon \in \{0.5, 0.01, 0.05, 0.001, 0.005, 0.0001\}$ の 24 種類のパラメータ計 2400 通りの組み合わせをそれぞれの方法で実行し, 近似誤差を計測した.

フィルタの繰り返し回数が 20 回 ($t = 20$) の場合の Peak Signal-to-Noise Ratio (PSNR) [23], [35] と最大誤差 E_{max} [23] の最大値を表 4 及び表 5 に示す. ここで, PSNR の計算では, Our-RGD にて $\epsilon \in \{0.5, 0.01\}$ のときに 1 件ずつ計 2 件 PSNR が無限大となったので, この無限大の数値を省いた平均を用いた. PSNR は値が大きい方が近似精度が高い. また, E_{max} は各画素の色誤差が全画素中で一番大きい誤差の値であり, E_{max} は低い方が近似精度が高い.

表 4, 表 5 の結果から Box 関数と Deriche を用いた場合

表 4: PSNR による近似精度比較.
Table 4 Accuracy comparison via PSNR.

ϵ	RG-GF	RG-GF-Deriche	Our-RGD
Total ave	37.8	58.1	287.2

表 5: 最大誤差による近似精度比較.

Table 5 Accuracy comparison via maximum error.

ϵ	RG-GF	RG-GF-Deriche	Our-RGD
Total max	8.40e-01	3.33e+03	7.81e-12

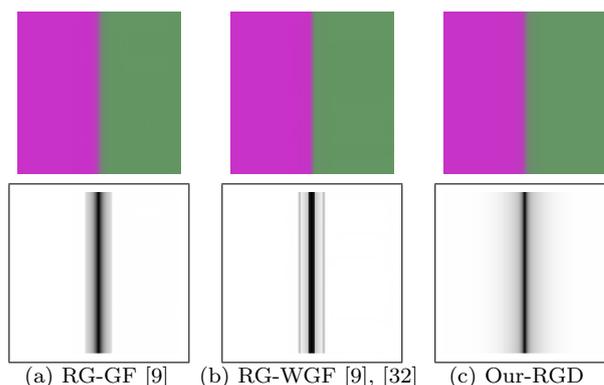


図 11: 偽エッジの有無 ($\sigma = 4$, $\epsilon = 0.04$). 既存法 (a,b) では不連続な勾配強度 (偽のエッジ) がアーティファクトとして下段画像にて確認できる.

Fig. 11 Comparison of RG filtering results (left) and their gradient magnitude (right) where $\sigma = 4$ and $\epsilon = 0.04$.

と比較して, DS 法では PSNR が高く, E_{max} が低いため, 高精度な近似結果が得られていることが分かる. 特に最大誤差 E_{max} では 7 桁~10 桁 Box 関数と Deriche 法より近似精度が高く, 提案法は超高精度と言える.

6.3 Visual Quality の比較

提案法と既存法の画質評価を顕著なエッジ周辺での偽エッジ有無の観点から行う. 図 9 に示した Step-edge, Mandrill, Lena の 3 枚を入力画像とし, 図 11-図 13 中の左はフィルタ結果, 右は対応する勾配強度画像であり, RG フィルタの繰り返し回数はいずれも 20 回 ($t = 20$) とした.

図 11 からいずれの既存法でも Box 関数の影響で, 提案法には無い偽のエッジ (入力画像には顕著なエッジが存在しない領域で勾配強度画像が不連続になる箇所など) が生成されていることが確認された. これは前節 6.2 で示した近似精度の定量評価やスケール空間理論からの予測 (ガウス関数と Box 関数での偽エッジの有無 [22]) と一致する結果である. RG-BF [9], [33] は提案法よりもエッジがシャープであるが, 図 12 より右側上部の髪とその隣接背景の境界部分に偽のエッジが確認される. RG-WGF [9], [32] も提案法よりはエッジがシャープであるが, 図 13 よりエッジが強調されすぎてしまい, 提案法には無い偽のエッジを生成していることが分かる.



図 12: RG-BF と提案法の品質比較: $\sigma = 16$, $\phi = 10.0$ (RG-BF), $\epsilon = 0.005$ (Our-RGD). 下段は上段に対応する拡大図.

Fig. 12 Quality comparison of the RG-BF and our filtering results (left) and their gradient magnitude (right) where $\sigma = 16$, $\phi = 10.0$ (RG-BF), $\epsilon = 0.005$ (Our-RGD).

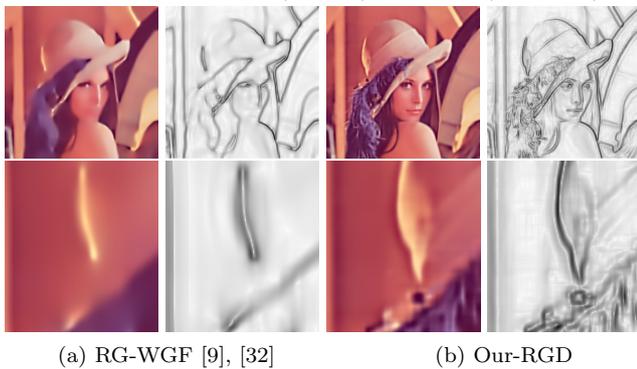


図 13: RG-WGF と提案法の品質比較: $\sigma = 8$, $\epsilon = 0.001$. 下段は上段に対応する拡大図.

Fig. 13 Quality comparison of the RG-WGF and our filtering results (left) and their gradient magnitude (right) where $\sigma = 8$ and $\epsilon = 0.001$.

6.4 収束性比較

文献 [13] で非一様な 1 次元畳み込み座標を分離実装することで生じる細長い構造での収束性の悪さを指摘されている。例えば RG-DT-DS [13] は図 14 及び図 15 から細長い構造では、繰り返し回数が増えるごとに徐々に収束していき一様でないことが分かる。一方、提案法 Our-RGD の結合フィルタは等間隔画素の計算であり、細長い構造でも収束度合いが同じであることが図 14 から確認できる。さらに、この一様性は以下で示す収束精度の実験結果から定量的に説明できる。

繰り返し回数毎に前後の 2 つの画像 (\mathbf{J}^t と \mathbf{J}^{t+1}) を用いた正規化 MAE と最大誤差 (全画素中最も大きなカラー差) D_{max} を計測した結果と対応するフィルタ後の画像例をそれぞれ図 16 と図 17 に示す。正規化 MAE のグラフを見ると全てのフィルタは繰り返し回数を重ねるごとに収束していることが分かる。しかし、RG-BF では他のフィルタと比べて収束が非常に遅いことが確認できる。また、 D_{max} のグラフを見ると、RG-BF を除いて一見収束しているように見えるが、RG-DT-DS [13] では、振動したりエ

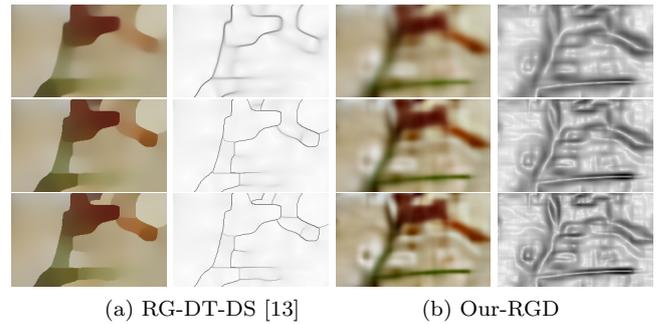


図 14: 図 15 の拡大図。RG-DT-DS は細長い領域の収束が一様ではないが、Our-RGD は一様に収束している。ただし、RG-DT-DS はよりエッジがシャープである。

Fig. 14 The close-up images of Fig. 15 where top to bottom correspond to the number of iterations such as $t \in \{4, 10, 20\}$, respectively.

ラーの値が高い。これは、RG-DT-DS の結合フィルタである定義域変換法が非一様な 1 次元画素を画像の行と列交互に (ジグザグに) 畳み込む分離実装であることに起因すると推測される。一方、提案法では図 16 で確認できるように高い収束性を得られた (他の入力画像・パラメータでも同様の結果を得ている)。これは、提案法では結合フィルタに等間隔画素の平滑化で構成される GF を用いて DS 法にて超高精度に計算している効果と考えられる。

7. おわりに

本稿では、高品質なフィルタ結果を効率的に生成するスケール対応画像フィルタ (Our-RGD) の計算法を新たに提案した。提案法では、スケール対応フィルタとして RG の結合フィルタを Ridge 回帰の再帰適用で構成した。また、初期平滑化と回帰計算内部の平滑化フィルタには、 L^1 ガウス関数を高速かつ超高精度に畳み込む DS 法を適用した。提案法を C++ プログラミング言語で実装し、その近似精度、計算速度、フィルタ結果の品質、及び収束度合いに関して先行研究の方法群と比較・評価した。その結果、既存法と同等の速度で近似精度及び品質が高く収束性が均一なフィルタ結果を生成することができた。特に、既存法では偽のエッジを生成するが多かったが、提案法ではそのようなアーティファクトは生成されない結果を効率的に得た。

一方で Our-RGD は RG-BF や RG-DT-DS 等の既存のスケール対応フィルタと同等の先鋭化度合いは得られなかった。それゆえ、正規化項の改良などがより良いフィルタ開発に必要と考える。また、勾配反転が少ない Ridge 回帰の特徴を生かして、本稿で提案した画像フィルタを特徴抽出、細部強調やスタイル画像化などの画像処理・解析問題へ応用することも今後の課題である。



図 15: RG-DT-DS (上段) と Our-RGD (下段) の比較結果. ここで, $t \in \{4, 10, 20\}$ (左から右), $\sigma = 16$, $\phi = 1.5$ (RG-DT-DS), $\epsilon = 0.001$ (Our-RGD) である.

Fig. 15 Quality comparison of the RG-DT-DS (top) and our (bottom) filtering result, where $t \in \{4, 10, 20\}$ (left to right), $\sigma = 16$, $\phi = 1.5$ (RG-DT-DS), and $\epsilon = 0.001$ (Our-RGD).

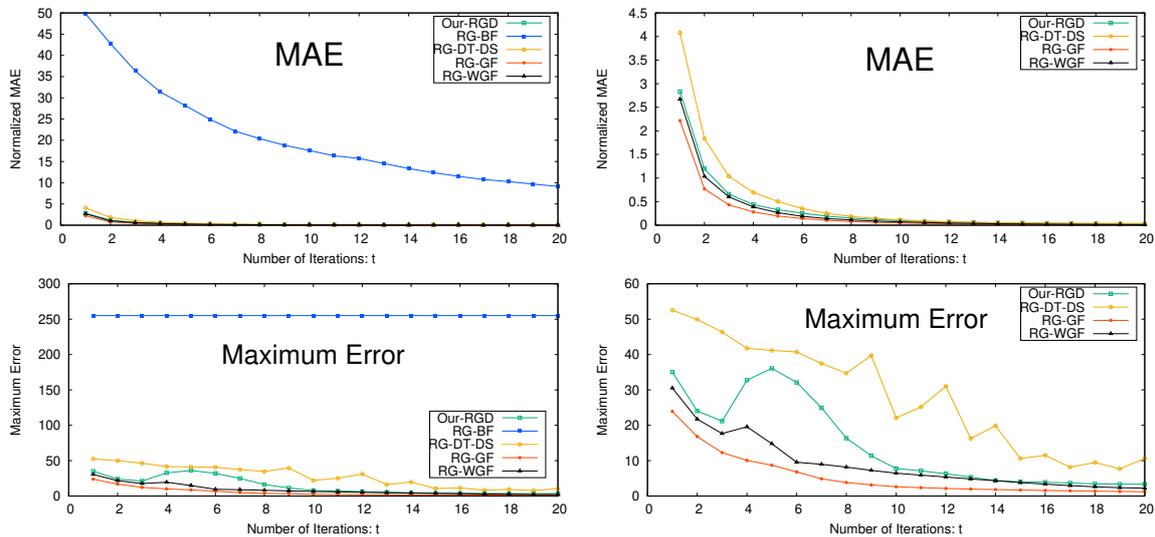
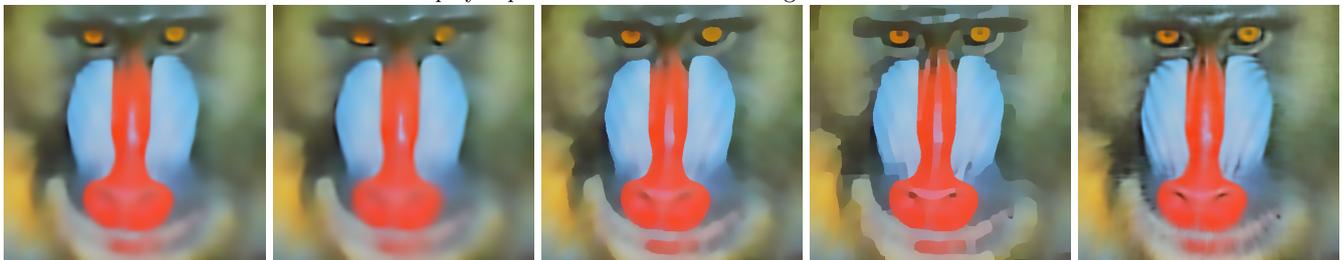


図 16: 収束性の比較. 上段と下段はそれぞれ繰り返し回数ごとの正規化 MAE 及び最大誤差のグラフである. 左図は今回用いたすべての既存法と我々のフィルタ結果であり, 右図は RG-BF を除いた結果である.

Fig. 16 Convergence comparisons with respect to the iteration numbers via the normalized MAE and maximum error (D_{max}) where the filtering results and their employed parameters are shown in Fig. 17.



(a) RG-GF [9] (b) RG-WGF [9], [32] (c) RG-BF [9], [33] (d) RG-DT-DS [13] (e) Our-RGD

図 17: 図 16 に対応するフィルタ結果: $\sigma = 8.0$, $t = 20$, $\epsilon = 0.01$ (RG-GF, RG-WGF, Our-RGD), $\sigma_r = 35$ (RG-BF), $\phi = 1.5$ (RG-DT-DS).

Fig. 17 Filtering results corresponding to Fig. 16 where $\sigma = 8.0$, $t = 20$, $\epsilon = 0.01$ (RG-GF, RG-WGF, Our-RGD), $\sigma_r = 35$ (RG-BF), and $\phi = 1.5$ (RG-DT-DS).

参考文献

- [1] Arbelaez, P., Maire, M., Fowlkes, C. and Malik, J.: Contour detection and hierarchical image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33, No. 5, pp. 898–916 (2010).
- [2] Felzenszwalb, P. F., Girshick, R. B., McAllester, D. and Ramanan, D.: Object detection with discriminatively trained part-based models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 9, pp. 1627–1645 (2009).
- [3] Yan, Q., Xu, L., Shi, J. and Jia, J.: Hierarchical saliency detection, *IEEE Computer Vision and Pattern Recognition*, pp. 1155–1162 (2013).
- [4] Gonzalez, R. C. and Woods, R. E.: *Digital image processing*, Upper Saddle River, N.J: Prentice Hall (2008).
- [5] Witkin, A.: Scale-space filtering, *International Joint Conference on Artificial Intelligence - Volume 2*, pp. 1019–1022 (1983).
- [6] Elder, J.: Are edges incomplete?, *International Journal of Computer Vision*, Vol. 34, No. 2, pp. 97–122 (1999).
- [7] Tomasi, C. and Manduchi, R.: Bilateral filtering for gray and color images, *International Conference on Computer Vision*, pp. 839–846 (1998).
- [8] Perona, P. and Malik, J.: Scale-space and edge detection using anisotropic diffusion, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 7, pp. 629–639 (1990).
- [9] Zhang, Q., Shen, X., Xu, L. and Jia, J.: Rolling guidance filter, *European Conference on Computer Vision*, pp. 815–830 (2014).
- [10] Kniefacz, P. and Kropatsch, W.: Smooth and iteratively restore: a simple and fast edge-preserving smoothing model, *CoRR*, Vol. abs/1505.06702 (2015).
- [11] Toet, A.: Alternating guided image filtering, *PeerJ Computer Science*, Vol. 2, p. e72 (2016).
- [12] Ham, B., Cho, M. and Ponce, J.: Robust guided image filtering using nonconvex potentials, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 1, pp. 192–207 (2017).
- [13] Yoshizawa, S. and Yokota, H.: Fast and faithful scale-aware image filters, *The Visual Computer*, Vol. 37, No. 12, pp. 3051–3062 (2021).
- [14] Sochen, N., Kimmel, R. and Malladi, R.: A general framework for low level vision, *IEEE Transactions on Image Processing*, Vol. 7, No. 3, pp. 310–318 (1998).
- [15] Frigo, M. and Johnson, S. G.: The design and implementation of FFTW3, *Proceedings of the IEEE*, Vol. 93, No. 2, pp. 216–231 (2005).
- [16] Deriche, R.: Recursively implementing the Gaussian and its derivatives, *IEEE International Conference on Image Processing*, pp. 263–267 (1992).
- [17] Greengard, L. and Strain, J.: The fast Gauss transform, *SIAM Journal on Scientific and Statistical Computing*, Vol. 12, No. 1, pp. 79–94 (1991).
- [18] Greengard, L. and Sun, X.: A new version of the fast Gauss transform, *Documenta Mathematica*, Vol. 3, pp. 575–584 (1998).
- [19] Getreuer, P.: A survey of Gaussian convolution algorithms, *Image Processing On Line*, Vol. 2013, pp. 286–310 (2013).
- [20] Brown, E. F.: Television: the subjective effects of filter ringing transients, *Journal of the Society of Motion Picture and Television Engineers*, Vol. 78, No. 4, pp. 249–255 (1969).
- [21] Bankman, I.: *Handbook of medical imaging*, Academic Press (2000).
- [22] Lindeberg, T.: On the axiomatic foundations of linear scale-space, *Gaussian Scale-Space Theory*, Springer, pp. 75–97 (1997).
- [23] Yoshizawa, S. and Yokota, H.: Fast L^1 Gaussian convolution via domain splitting, *IEEE International Conference on Image Processing*, pp. 2908–2912 (2014).
- [24] Gastal, E. and Oliveira, M.: Domain transform for edge-aware image and video processing, *ACM Transactions on Graphics*, Vol. 30, No. 4, pp. 69:1–69:12 (2011).
- [25] He, K., Sun, J. and Tang, X.: Guided image filtering, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 6, pp. 1397–1409 (2012).
- [26] Petschnigg, G., Szeliski, R., Agrawala, M., Cohen, M., Hoppe, H. and Toyama, K.: Digital photography with flash and no-flash image pairs, *ACM Transactions on Graphics*, Vol. 23, No. 3, pp. 664–672 (2004).
- [27] Eisemann, E. and Durand, F.: Flash photography enhancement via intrinsic relighting, *ACM Transactions on Graphics*, Vol. 23, No. 3, pp. 673–678 (2004).
- [28] Paris, S.: A gentle introduction to bilateral filtering and its applications, *ACM SIGGRAPH 2007 courses*, ACM, pp. 3–es (2007).
- [29] Draper, N. R. and Smith, H.: *Applied regression analysis*, Vol. 326, John Wiley & Sons (1998).
- [30] Hastie, T., Tibshirani, R., Friedman, J. H. and Friedman, J. H.: *The elements of statistical learning: data mining, inference, and prediction*, Vol. 2, Springer (2009).
- [31] Bashkirova, D., Yoshizawa, S., Latypov, R. and Yokota, H.: Fast L^1 Gauss transforms for edge-aware image filtering, *Proceedings of Institute for System Programming of the Russian Academy of Sciences*, Vol. 29, No. 4, pp. 55–72 (2017).
- [32] Li, Z., Zheng, J., Zhu, Z., Yao, W. and Wu, S.: Weighted guided image filtering, *IEEE Transactions on Image Processing*, Vol. 24, No. 1, pp. 120–129 (2014).
- [33] Adams, A., Baek, J. and Davis, M. A.: Fast high-dimensional filtering using the permutohedral lattice, *Computer Graphics Forum*, Vol. 29, No. 2, pp. 753–762 (2010).
- [34] Wells, W. M.: Efficient synthesis of Gaussian filters by cascaded uniform filters, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 2, pp. 234–239 (1986).
- [35] Paris, S. and Durand, F.: A fast approximation of the bilateral filter using a signal processing approach, *European Conference on Computer Vision*, pp. 568–580 (2006).