

顕著度マップによる基礎的な画像処理アルゴリズムについての研究

**Research on the Basic Image Algorithms by Using
Conspicuousness Map**

盛 磊

主 査: 張 善 俊 教授
副 査: 中田 穰治 教授
副 査: 桑原 恒夫 教授
副 査: 内田 啓一郎教授
副 査: 中山 堯 教授

〈あらまし〉 本論文は人間の視覚特性に着目し、顕著度マップを導入することで従来よく使われている基礎的な画像処理のアルゴリズムを研究し、新しいモルフォロジー処理と新しい画像の拡大縮小のアルゴリズムを提案している。本論文では、画像中の画素の「目立つ」度合を顕著度マップとして定義し、従来のモルフォロジー処理に必要な画素間の大小順序を画素の値より固定しておくやり方と違って、画像の中身に応じては画素間の「目立つ度合い」を利用して動的にその大小順序を決める方法を提案した。結果的に従来手法のような画像の中身によって処理効果が変わることが少なくなり、従来のモルフォロジー処理で得られなかった安定的な処理効果を実現した。この方法は複雑な色環境における大きいカラーノイズをうまく除去することに成功した。また、画像応用の普及によって、高精度の画像拡大／縮小のニーズが大きい。本論文は顕著度マップを利用して、高倍率の画像縮小処理に特化した補間法を提案した。従来の固定的なサンプリング範囲を持つ補間法は高倍率の縮小処理において、サンプリング間の空間が空けられ、処理された結果画像は不自然さが残る。一方、面積平均法を使う従来法はあらゆる画素の値を均一に処理に加えるため、逆に画像の特徴を薄くしてしまう結果になる。提案手法は高倍率の画像縮小過程において、元画像の特徴点を失うことを抑えつつ、結果画像における間引き問題を回避した。実験で提案手法は従来のアルゴリズムを改善したことを実証した。

キーワード：顕著度マップ カラー画像 モルフォロジー処理 補間法

〈Summary〉 A new concept of conspicuousness map is introduced and applied in the basic image processing algorithms according to the human visual sensibility. In this paper, a new morphological color image processing algorithm and a new scaling interpolation algorithm have been explained in detail. The conspicuousness map is defined on the HSV color model for every pixel in the image. It works better than the traditional edge detection method to show the local difference of a pixel in the image. In the new morphological processing algorithm, the pixels order needed in the erosion and dilation processing is not decided by the value of the Euclidean distances between the RGB vector of the pixels, but by the conspicuity of the pixels according to the neighbor pixels dynamically. The efficiency of the new morphological algorithm has been proved by the experiments, where noise in the complicated color image can be removed properly. As another application of the conspicuousness map, a new scaling interpolation method is used for controlling the area average calculation for the resize of the original image, especially for great scaling zoom out ratio of the image. Traditionally, when a digital bitmap image is zoomed out, a pixel in the result image may correspond to a set of pixels, it may be effaced by the surrounding pixels. If the pixel contains the important image feature, say, it is the pixel with high conspicuity; the resulting image may lose the important image information. In the proposing algorithm a balance between the smoothness and sharpness of the processing is under consideration. Experimental results show the advantages of the proposing method.

Keywords: conspicuousness map, color image, morphology processing, interpolation.

目次

1 総論	1
2 顕著度マップ	3
3 顕著度マップによるモルフォロジー処理.....	10
3.1 背景.....	10
3.2 従来のモルフォロジー処理法	11
3.3 顕著度マップによるモルフォロジー処理	13
3.4 実験及び従来法との比較	14
4 顕著度マップによる補間法.....	21
4.1 背景.....	21
4.2 従来の補間法.....	21
4.3 画素のローカル投影面積	22
4.4 顕著度マップを利用した面積平均法	26
4.5 実験及び従来法との比較	27
5 まとめ	36
参考文献	37
発表リスト	38

図 2.1	保護色	3
図 2.2	航空障害灯	4
図 2.3	画素値差分の比較	4
図 2.4	エッジ検出	6
図 2.5	変色円環を含む画像	6
図 2.6	白黒画像の顕著度	8
図 2.7	顕著度マップ	9
図 3.1	グレースケール画像のモルフォロジー処理	12
図 3.2	チャンネル別の膨張処理	12
図 3.3	モルフォロジー処理例	13
図 3.4	円環除去	15
図 3.5	ホワイトノイズの除去	16
図 3.6	カラーノイズ画像	16
図 3.7	カラーノイズの除去	17
図 3.8	収縮処理	18
図 3.9	ノイズ除去の効果の比較	19
図 3.10	円環除去の効果の比較	19
図 4.1	目標画像による原画像の領域分割	23
図 4.2	投影範囲	24
図 4.3	拡大処理による可能な投影	25
図 4.4	風景画像による縮小処理	28
図 4.5	草の部分の拡大図	29
図 4.6	投影範囲の間の空白領域	30
図 4.7	樹の部分の拡大図	31
図 4.8	人造画像による縮小処理	35

1. 総論

計算機と電子製品の発展および普及により、デジタル画像技術は現代社会の隅々まで深く浸透してきている。ナビゲーション、カメラ、画像処理ソフトに代表される多彩な生活用品は増え続けている。経済活動や工業生産においても、通信、ビデオ、生物、測量と製図など多数の領域にあたって、デジタル画像技術は重要な役割を果たしている。また、レーダー、航空、ディープスペース探査を始め、最先端の軍事、科学研究の分野においてもデジタル画像技術は必要不可欠な技術となっている。具体的に、リモートセンシングによる地形情報、山火情報、農地情報の集取、転送、分析、生物医学による顕微、超音波画像、通信による圧縮、解凍、暗号化、工業生産における品質検査、軍事分野におけるナビゲーション、地形マッチング、文化産業における絵やアニメの創作など、様々な画像処理技術が社会の発展に貢献している。

様々な分野で使われているデジタル画像技術は大きく三種類に分類される。つまり、画像処理、画像分析と画像理解である。画像処理技術は、主に画像の収集、保存、再現、変換、フィルタリングおよび修復技術である。画像分析は、主にエッジ検出、セグメンテーション、形状分析、運動識別などである。画像理解は、マッチングや画像検索などである。

デジタル画像処理は1964年JPL(Jet Propulsion Laboratory)によって最初に実用され、レインジャー7号が月から送られた写真を使い、月の表面地図の制作に成功した。また、1972年CTの発明による、画像処理の応用領域は大幅に拡張した。次々の成功と計算機性能の発展により、伝統的なアナログベースの光学画像処理の領域はだんだんデジタル画像処理に侵食され、現在はデジタル画像処理の方が主流となった。さらに、画像処理技術から生まれたコンピュータグラフィックス技術も新たな独立分野となり異彩を放っている。

デジタル画像処理の発展において、基盤である基礎的な画像処理アルゴリズムの研究は重要である。ほとんどの画像処理プロジェクトは汎用性が高い基礎的アルゴリズムを使っている。基礎的アルゴリズムの性能は多くのアプリケーションの性能を左右している。そのため、基礎的アルゴリズムを研究し改善していくことがとても大事である。

ノイズを除去する要望や画像を拡大／縮小する要望は画像処理の様々な場面で出てくる。本論文は画像の「目立つ」部分の検出に対して、顕著度マップを利用した新たなアルゴリズムを提案する。顕著度マップは人間の視覚の感性を優先し、HSV(色相、彩度、明度)モデルにおける画素と周囲画素と違いを人間の視覚感覚とより一致するように表現するマトリックスである。本論文は顕著度マップを使い、画像の形態学的なフィルター処理を二値画像、濃淡グレースケール画像およびカラー画像において統一できるよう定義し直した。また、離散的デジタル画像のスケーリング処理における補間方法を汎用性を失わずに改善した。

本論文の構成は次の通りである。第二章は顕著度マップの概念を導入し、人間の視覚感覚に合うように画像中の目立つ部分を数値的に示すマトリックスの構築方法を説明する。次に、第三章は顕著度マップによりモルフォロジー処理法を提出し、その有効性及び優越性を実験で示す。続いて、第四章は顕著度マップによる面積平均法を提案し、従来の拡大／縮小のスケーリングアルゴリズムを改善した。最後に第五章でそれらのアルゴリズムの特徴点をまとめ、残されている問題と将来の課題について述べる。

2. 顕著度マップ

脳は目立つものに対して注目しやすいといわれている。自然界では、多くの生物が本能的にその原理に準じて、長い時間を経て進化して保護色を得た。例として、図 2.1 の葉の上に止まっている虫は自分の色を周りの環境の色に近づくことで、天敵などの捕食者から身を守るようにしている。我々は一目でその虫を簡単に見つけられないというのは、おそらく脳が普段通りの平均的な処理で、虫の存在を検出できないと考えられる。しかし、もし事前にそのあたりに虫があると教えておいてから人に虫を探させるなら、ほとんどの場合その虫を見つけることができる。その理由は、日常的に膨大な視覚刺激(画像情報)は絶え間なく我々の目に入ってきているが、我々の脳はそれらの情報をすべて同じように計算資源を振り分けて処理しているわけではない。我々の脳は関心を持つ情報を重点的に処理し、それ以外のものを少ない計算でより簡単に処理をやっていると思われる。我々は画像を観察するとき、緩やかに変化する周囲から急激に変化する画像部分は私たちの注意を引きやすい。画像において、ある画素は周りの画素との違いが高いほど、その画素は注意されて処理される可能性が高くなる。これは日常の生活から体験できる。例えば、図 2.2 に示す夜景の中に多くの建物は映っているが、多くの人は建物の中の細かい部分よりも赤い航空障害灯に注目すると思われる。

画像中の画素の変化を検出するのに、従来は Sobel フィルター^[1]やラプラシアンフィルター^[2]などのエッジフィルターが利用されてきた。RGB の三チャンネルのエッジ処理結果で得られている画素の数値的变化は視覚的感ずる変化とは必ずしも一致していない。例えば、図 2.3 の円に示されている6つの色は時計順から見てそれぞれ水色(0, 255, 255)、緑色(0, 255, 0)、黄色(255, 255, 0)、赤色(255, 0, 0)、紫色(255, 0, 255)と青色(0, 0, 255)である。数値的に考えると、それぞれの

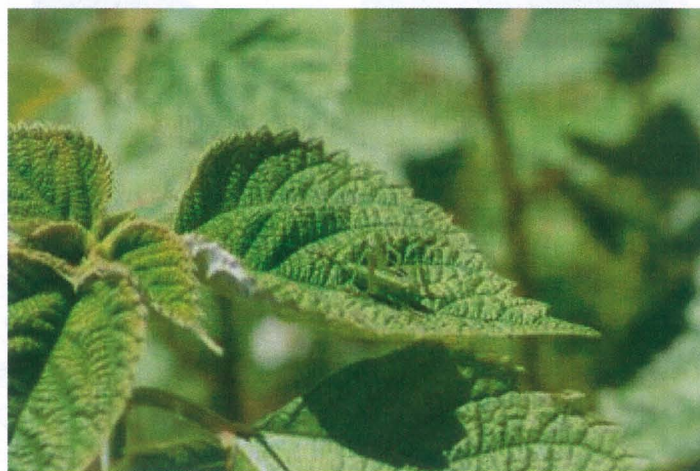


図 2.1 保護色

fig. 2.1 Protective coloration



図 2.2 航空障害灯

fig. 2.2 Aviation obstacle lights

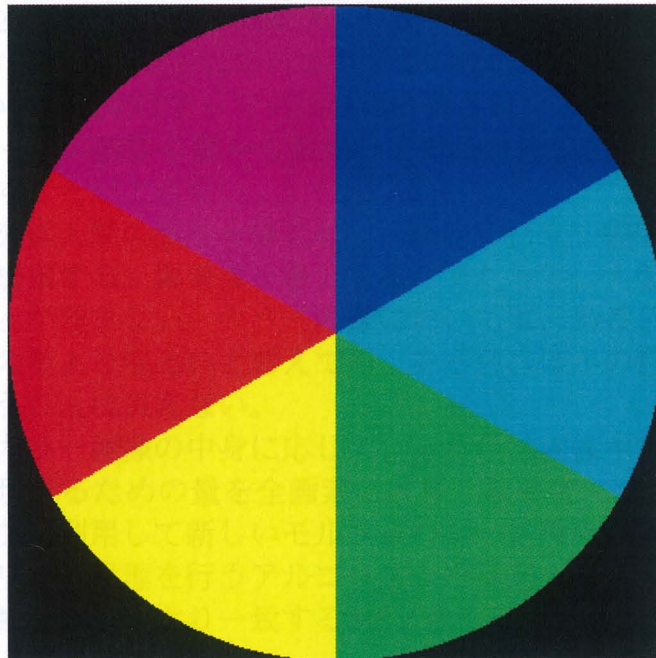


図 2.3 画素値差分の比較

fig. 2.3 Comparison about the difference values of pixels

色ベクトルとその隣接の色ベクトルの間の数値差のノルムは全部 255 で同じになっている。しかし、視覚的な感覚から青色から紫色までの色変化と赤色から黄色までの色変化量は同じではない。

人間の視覚特性に着目し、従来の RGB(赤、緑、青)色モデルにおける問題点を説明する。図 2.4(A)は 3つの単純色の背景にそれぞれ 3 個の色違いの正方形の色領域を置くことにより構成されている画像である。左から右への背景の色を RGB で表すとはそれぞれ $(128, 0, 0)$ 、 $(0, 128, 0)$ と $(0, 0, 128)$ で、赤、緑と青となっている。前景の正方形は背景の色から変化を加えて設定している。上段、中段、下段はそれぞれ三つの正方形がある。それらの正方形の色は同じ段で同じ規則に沿って変化している。

一番上の段の色ベクトルは値がゼロとなっているあるチャンネルの値を 40 にして、それぞれの色ベクトルを(128, 0, 40)、(0, 128, 40) と(0, 40, 128) に設定する。このように設置した前景の正方形の色ベクトルと背景の色ベクトルとの差分のノルムは同じであるが、人間が感じる前景と背景の色違いの程度は違う。

中央の段の三つの正方形の色は左から右へそれぞれ $(128, \sqrt{40}, \sqrt{40})$ $(\sqrt{40}, 128, \sqrt{40})$ と $(\sqrt{40}, \sqrt{40}, 128)$ に設定している。中央の段の前景と背景の色ベクトルの差分のノルムも同じく 40 となっているが、人間が感じる前景と背景の色違いの程度は同じ段で違うし、上段のそれと比べても人間が感じる色違いの程度が違う。

下段では、正方形の色は(128-40, 0, 0)、(0, 128-40, 0)と(0, 0, 128-40)に設定している。この段の前景と背景の色ベクトルの差分のノルムも同じ 40 となっているが、人間が感じる前景と背景の色違いの程度は違う。

図 2.4(B)は図 2.4(A)の画像を Sobel フィルターに適用して検出されたエッジ画像である。図 2.4(B)から検出されているすべての正方形のエッジのノルムは同じ値 40 となっている。従来はエッジの強さは計算されたエッジのノルムで現しているが、このエッジの強さの値から前景のどの正方形の色は背景の色より目立って違うかを示すことができない。

そこで、本研究は人間の感性を優先にし、画素の目立つ程度を表現する顕著度という概念を提案する。顕著度を導入するにあたって、人間の直感に近い数値で定義することが望ましい。カラー画像においては、ある色は別の色より濃い、あるいは別の色より明るいと言えども、ある色はほかの色より大きいあるいは小さいということはいえない。

本論文では、カラー画像の中身に応じて、ある色は画像中でどれほど目立つかを数値的に表現するための量を全画素で計算し顕著度マップとして定義する。この顕著度マップを利用して新しいモルフォロジー処理のアルゴリズムと新しい画像のスケーリング処理を行うアルゴリズムを提案する。

以下は人間の視覚感覚とより一致する HSV 色モデルに基づいて顕著度マップを説明する。

HSV 色モデルは画像の色相 H(Hue)、彩度 S(Saturation)、明度 V(value)を数値的に表記する色モデルである。色相は色の大きな色の種類を表し、彩度は色の鮮やかさを表し、明度は色の明るさを表している。

図 2.5 の円における円周方向の色の種類の変化がパラメータ H によるものである。半径方向色の純度の変化はパラメータ S によるものである。さらに、パラメータ V でそれらの色の輝度を調整することはできる。HSV モデルと人間が色を知覚する方法と類似しているため、本論文では、HSV の値を利用して、画像中のある画素の目立つ度合を算出するようにしている。

式 2.1 は図 2.5 の円における点の色の (H,S,V) の値を決めるための式である。式 2.2 は図 2.5 の中にある円環の色を決めるための式である。w と h は画像の大きさで、x, y は円の中の点の座標である。式 2.2 で決めている円環の太さは背景の大円の半径の長さの 2%である。

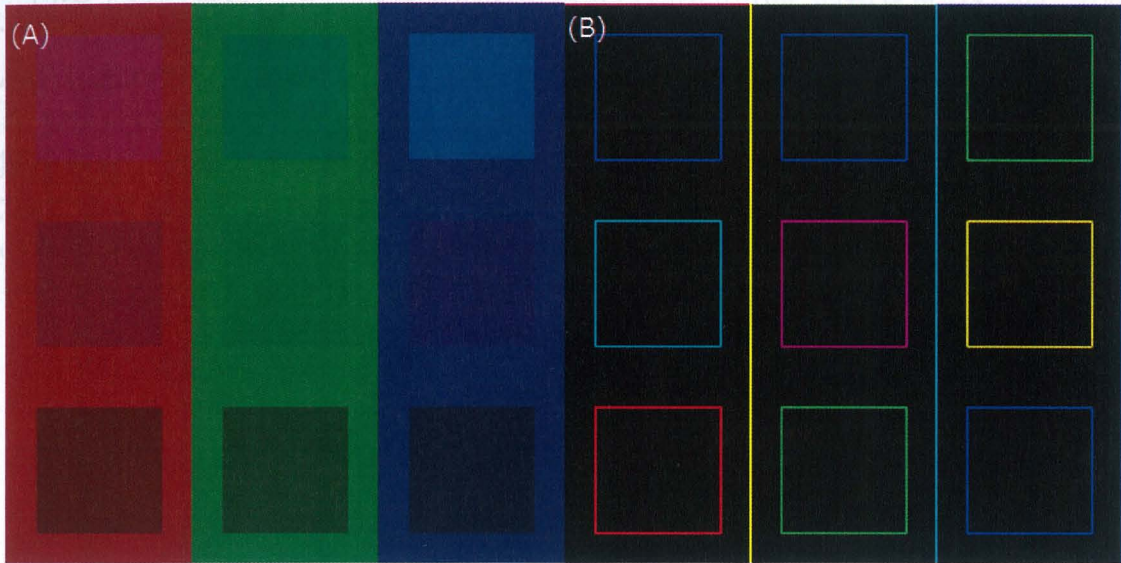


図 2.4 エッジ検出
fig. 2.4 edge detection

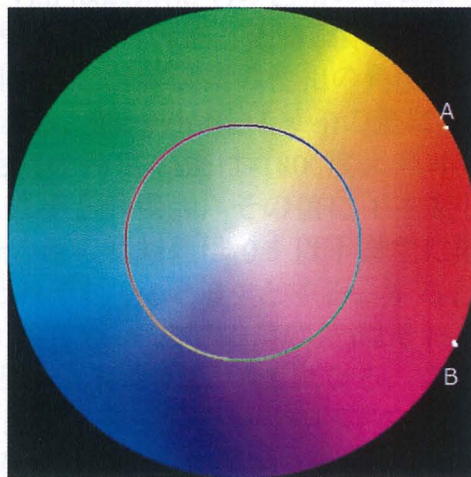


図 2.5 変色円環を含む画像
fig. 2.5 A color image with varying circular ring

$$\begin{cases} \tan H_1 = \frac{h/2 - y}{x - w/2} \\ S_1 = \sqrt{\frac{(h/2 - y)^2 + (x - w/2)^2}{(h/2)^2 + (w/2)^2}} \\ V_1 = 1.0 \end{cases} \quad (2.1)$$

$$\begin{cases} H_2 = (H_1 + 180) \% 360 \\ S_2 = 1.0 \\ V_2 = 1.0 \end{cases}, \text{if } |S_1 - 0.5| < 0.01 \quad (2.2)$$

円環中の画素の色相 H は背景にある画素の色相を 180° 回転し得られる色相値である。円環の彩度 S を 1.0 に設定しているが、背景円上の点の彩度 S は円の中心から離れている距離によって決められている。ただし、色相 H の値はゼロから 360 度の範囲内である。ゼロ度と 360 度は同じ色相を意味する。色相値の周期性から、色相の差分計算をする時には特別な配慮が必要である。例えば、図 2.5 にある点 A と点 B の色はほんの少ししか変化していないが、その間の色相 H の差は 180 度を超えている。色相の差分の大きさを正しく色の違いの大きさと一致させるため、色相 H の差分は式 2.3 を持って修正する。

$$t(x, y, i, j) = \begin{cases} -1, & f(x, y).h - f(x+i, y+j).h > 0.5 \\ 1, & f(x, y).h - f(x+i, y+j).h < -0.5 \\ 0, & \text{others} \end{cases} \quad (2.3)$$

(x, y) は注目画素の位置であり、(i, j) は注目画素を中心とする構造要素内で処理に参加している画素の相対位置である。

式 2.3 において、 $f(x, y).h$ は画素(x, y)の色相のゼロから 360 度の角度をゼロから 1.0 に変換した値を表す。注目画素と近隣画素(x+i, y+j)との色相の差が 0.5 以上の場合、つまり、注目画素と近隣画素の色相の角度の差は 180 度以上の場合、その角度差を対角に修正し、 $f(x, y).h - f(x+i, y+j).h$ の差に -1.0 を足す。また、もし注目画素と近隣画素の色相の角度の差は -180 度より小さい場合、つまり、 $f(x, y).h - f(x+i, y+j).h$ は -0.5 より小さい場合、その値に 1 を足して対角に変換する。その結果、色相 H の差分の値の絶対値は 0.5 以下までに限定される。例えば図(3)の A 点と B 点の H をそれぞれ 0.1 と 0.9 とする。すると、 \overline{AB} (点 A から点 B までの角度差) と \overline{BA} (点 B から点 A までの角度差) はそれぞれ -0.8, 0.8 となり、補正関数 t で修正を行うと、結果は 0.2, -0.2 になる。この補正関数は点 A とその時計回りに 180° 以内の点との角度の差を 0.5 以下の正数にし、逆時計回りに 180° 以内の場合は -0.5 までの負数にする効果がある。

次に、それを利用して感性的な「目立つ」を数値化する方法を述べる。白黒二値画像、多値グレースケール画像とカラー画像において検討し、最後にそれらを統一させる。

図 2.6 は 3×3 サイズの白黒画像である。中央画素を注目画素とする。注目画素は 3×3 の構造要素内においてどれくらい「目立つ」であるかは注目画素と色違いの画素の数をカウントすることで表すことができる。

多値グレースケール画像の倍意は注目画素目立つ度合いをその画素とそれを中心とする構造要素内の別画素との画素値の差を総和を取って現す。

カラー画像の場合は、画素の色ベクトルをまずいったん HSV 表現に直す。そして、注目画素と注目画素を中心とする構造要素内のそれぞれの近隣画素の H, S, V の成分と差分を計算する。色相の差分は式 2.3 で修正してからそれぞれの成分の総和を計算する。式 2.4 は n×n の構造要素における注目画素の相対的色変換を求める式である。この式の計算で得られたベクトルは顕著度ベクトルと呼ぶ。

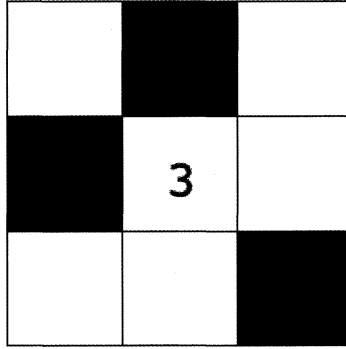


図 2.6 白黒画像の顕著度

fig. 2.6 Conspicuousness degree for black and white image

$$\overline{c(x, y)} = \sum_{\substack{i=-n, j=-n \\ i=n, j=n}} \{ \overline{f(x, y)} - \overline{f(x+i, y+j)} + (t(x, y, i, j), 0, 0) \} \quad (2.4)$$

式 2.4 では、 $\overline{f(x, y)}$ は画素(x, y)の色ベクトル(画素値)である。そのベクトルは HSV 色空間によるものであるため、ベクトルの要素は色相、彩度と明度の値になる。n は構造要素の大きさと形状によって変わるが、本論文は正方形の構造要素を使う。注目画素と構造要素内のすべての画素との差分から、各々の画素の目立つ程度を示す顕著度ベクトル $\overline{c(x, y)}$ を求める。構造要素はモルフォロジー処理をする際に、注目画素を含むある形を持つ領域のことである。通常は注目画素を中心とする $n \times n$ の正方形であるが、特殊な形や大きさを選ぶ場合もあるので、この構造要素の形や大きさの違いによって顕著度の値の解釈への影響を抑えるため、式 2.5 で顕著度ベクトルの正規化を行う。

$$\overline{c_s(x, y)} = \frac{\overline{c(x, y)}}{m-1} \quad (2.5)$$

式 2.5 中の C_s は正規化された顕著度であり、パラメータ m は構造要素内に含まれる画素の数である。通常の構造要素は注目画素自身を含めているので、注目画素とは別の画素の平均的な違いを求めるときは、注目画素を取り除く必要があるから分母の数をひとつ引くようにして構造要素内の画素の総数から注目画素自身を減らすとした。注目画素が画像の縁にあると構造要素が画像の範囲外に延びた場合、範囲外の画素を式 2.4 の計算に参加させないで、式 2.5 の正規化処理をする際にも、割り算の分母は式 2.4 の計算に加担した注目画素以外の画像構造要素内の画素の数とする。以上の計算で得た結果の色相に対応する $\overline{c_s(x, y).h}$ の値の範囲は[-0.5, 0.5]で、彩度に対応する $\overline{c_s(x, y).s}$ と明度に対応する $\overline{c_s(x, y).v}$ の値の範囲は同じ[-1.0, 1.0]である。

顕著度の保存、表示および GPU を使った高速化などプログラミング上の都合により、全画素の正規化顕著度の値の範囲を式(2.6)でゼロから 1 の間にシフトさ

せる。そうすると顕著度ベクトルの三成分は[0,1]の小数となり、画像として解釈したり、表示したりすることが便利となるし、GPU での高速プログラム処理をする際にも便利となる。

$$\overline{c_r(x,y)} = \frac{\overline{c_s(x,y)} \times (2,1,1) + (1,1,1)}{2} \quad (2.6)$$

式 2.6 の $\overline{c_r(x,y)}$ により構成される画像は本論文では、顕著度マップと定義する。

図 2.4(A)の画像から求めた顕著度マップは図 2.7 になる。図 2.7 において、図 2.4(B)の色差分により、顕著度マップは各ブロックと背景の色違いをはっきり示した。この図から、下部の明度が違い部分は最も顕著になっていることが分かる。色の種類の違いより色の鮮やかさの違いをより顕著度に現している定義となっている。心理学的に、人間は赤色に注意が喚起しやすいが、赤色の中での変化を人間は気づきづらいことが分かっている。また緑の色違いの変化は人間が敏感に察知できるということがあるが、今回はこれらの事実を顕著度マップの計算に反映されていない。また、本論文で顕著度マップを定義する際に、HSV のそれぞれの成分に重みを加味していないが、今後 HSV それぞれの影響ファクターを検討した上でより人間の感覚にマッチするような顕著度を定義できると考えている。

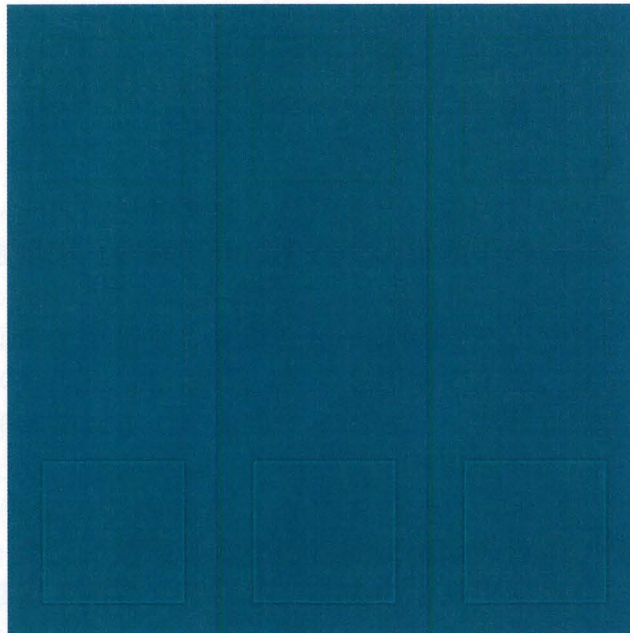


図 2.7 顕著度マップ

fig. 2.7 Conspicuousness map of color image

3. 顕著度マップによるモルフォロジー処理

3.1 背景

モルフォロジーフィルタ処理は構造要素 (SE) における非線形信号整列技術である。最初に Matheron と Serra によって開発されたマスマティカル・モルフォロジー技術は画像処理とコンピュータビジョンに広く利用されている^{[1][4]}。膨張 (Dilation) と収縮 (Erosion) 処理はモルフォロジー処理の基本的なオペレーターであり、それらの組み合わせであるオープニング (Opening) 処理、クロージング (Closing) 処理、ヒット (Hit) 処理、トップハット (Top-Hat) 処理などはノイズ除去、画像強調、特徴抽出、テクスチャ分析に応用されている^{[5]-[10]}。

モルフォロジー処理は、一般的に 2 値画像に対して行われるが、濃淡画像やカラー画像にも応用可能である。白黒画像の平滑化、ノイズ除去と小さな穴や凹凸の除去などの画像処理にモルフォロジー処理がその威力を発揮している。グレースケール画像およびカラー画像では、注目画素の値をその画素を中心とするある構造要素中の画素集合を集合演算によって決定する。

例えば、入力信号 $f(n)$ の集合 $F=\{f_1, f_2, \dots, f_n\}$ 、構造要素 $g(m)$ の集合 $g(m)=\{g_1, g_2, \dots, g_n\}$ があるとすると、膨張処理 \oplus と収縮処理 \ominus はそれぞれ式 3.1 と式 3.2 で定義される^[7]。

$$(f \oplus g)(n) = \max \left\{ \begin{array}{l} f(n-m) + g(m) \\ 0 \leq n-m \leq n, m \geq 0 \end{array} \right\} \quad (3.1)$$

$$(f \ominus g)(n) = \min \left\{ \begin{array}{l} f(n+m) - g(m) \\ 0 \leq n+m \leq n, m \geq 0 \end{array} \right\} \quad (3.2)$$

白黒画像やグレースケール画像などスカラー値のデータ集合を対象とする場合において、膨張処理と収縮処理はすべての画素の値をそれぞれの構造の中で最も大きい(小さい)値にする。基本的に、この処理で得られる結果画像中は元画像に存在しない画素値を発生させることがない^{[5][6][9]}。一方、カラー画像の場合は、画素の値はスカラーではなく、ベクトル型である。従来のモルフォロジー処理をカラー画像の領域までに適用するのに、ベクトル型の画素値の間の順序を決める必要がある。過去では、ユークリッド距離やベクトル間の内積などを用いて、カラー画像の画素値の大きさを規定する研究が多数行われている^{[9][10][11]}。モルフォロジー処理の本来の目的は画像内のオブジェクトの形状をある目的に応じて整形することである。従来のモルフォロジー処理は数学的な視点から決められたベクトル間の大小関係を注目画素と構造要素内の別の画素間で調べ、最大あるいは最小の画素位置を選び、その位置にある画素を膨張あるいは収縮の結果画素とした。しかし、違う画像内のオブジェクトの形は同じでも、オブジェクト周囲の画素値ベクトルの値の如何によって、同じモルフォロ

ジー処理の結果として、画像内のオブジェクトの形の変化は異なってしまうことがよくある。処理結果を予想とおりにすることが困難な場合がある。画像内のオブジェクトの形は周囲の画素との相対的な色の違いによって認知されている。大小関係の判断が重要なモルフォロジー処理においてある画素の色の値がほか画素の色の値より大きいということを視覚心理学的に決めることは難しい。事前に色の大小関係を決めておいても、ある画像の処理に通用しても、別な画像の処理では不都合が生ずる。画像の内容に応じて動的に画像間の大小順序を矛盾が発生しないように決める方法が望まれる。また、単純に画像の各色チャンネルに従来のグレースケール画像のモルフォロジー処理を別々に処理してから統合すると、元画像に存在しない色が発生してしまう。具体的例は後述する。

本章は構造要素中の画素間の大小順序を人間が画像を見る際の感覚に合うように、画素間の感性的な類似程度ある程度示せる顕著度を利用する。白黒二値画像も、濃淡画像もカラー画像も、統一的に取り扱うモルフォロジー処理を本章で詳しく説明する。また、本論文で提案する新しいモルフォロジーのアルゴリズムは従来法では難しかったカラー画像処理の効果が得られた。3.2節では従来のモルフォロジー処理法を紹介した上で、3.3節では顕著度マップを使った新手法を説明する。3.4節では、比較実験を持って提案アルゴリズムの優位性を示す。

3.2 従来のモルフォロジー処理法

従来のモルフォロジー処理は、画像内容に関係なく、あらかじめ決めた画素の大小順序を用いて処理を行う。このやり方では、開発者が考えた特殊な応用対象によい結果が出る反面、他の一部の対象画像に逆効果が出る場合がある。

図 3.1(A)は 8bit のグレースケール画像である。画像の中央部分を除いて、画素の値は上から下まで徐々に上昇するとしている。画像の中央部の細い帯状の画像部分の画素値は逆に上から下へ減少させている。グラデーションで変化する画素を背景画素として、帯状の画素を前景のオブジェクトと見ることができる。伝統的な膨張処理と収縮処理で図 3.1(A)にある帯状の画像部分を除去・軽減しようとする、その帯状の画像部分は図 3.1(B)、図 3.1(C)が示すように、帯状のオブジェクトの上と下の部分は全く逆の結果になってしまう。収縮処理の場合、注目画素はその構造要素の中の最も小さい値の画素値で置き換えられ、図 3.1(B)のように上部は細くなる。それに対して、下の部分の画素値は両端より小さいため、その形状は太くなる。同じ理由で、図 3.1(C)の現象も説明できる。図 3.1 が示すように、一定な画素値分布を持つ画像において、伝統的モルフォロジーの処理効果は限られている。

カラー画像の領域までに拡張すると、問題はさらに深刻になる。通常の 24bit のカラー画像は 3 つのチャンネルで構成されたものであるため、スカラーの画素値を処理対象とする伝統的なモルフォロジー処理をベクトル型の処理対象へ拡張する必要がある。図 3.2(A)は大きさ 3×3 のカラー画像とする。中心画素の黒色の周りに、青色、緑色、水色、紫色、赤色が置いてある。中心の画素に対して、 3×3 の構造要素で膨張処理を行うと、注目画素の各チャンネルはそれぞれ

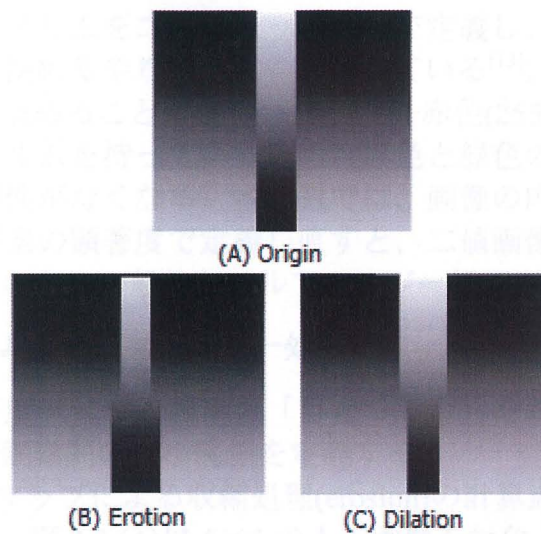


図 3.1 グレースケール画像のモルフォロジー処理
 fig. 3.1 Morphology process of gray scale image

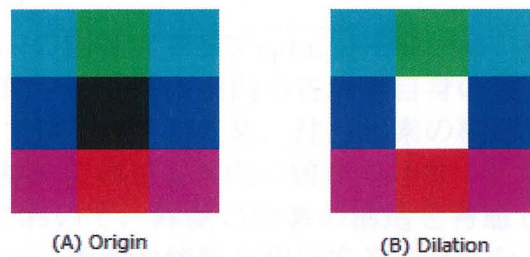


図 3.2 チャンネル別の膨張処理
 fig. 3.2 Dilation process of different color channels

周りの対応チャンネルの最大値で置き換えられる。従来のグレースケール画像の膨張処理を単純にカラー画像の各チャンネルにそれぞれ適応すると、図 3.2(B)に示されるような結果になる。中央の黒の画素を注目する。注目画素を中心とする 3×3 の構造要素内の画素は水色(0, 255, 255)、赤色(255, 0, 0)青(0,0,255)、緑(0, 255,0)と紫(255,0,255)となっている。従来の膨張処理を注目画素のそれぞれのチャンネルに適用すると、注目画素の各チャンネルの処理結果は構造要素内の画素のそれぞれの値の最大値の 255 となり、結果的に元画像に存在しない白色(255, 255, 255)になってしまう。一般的に、モルフォロジー処理は新たな色の発生を避けるべきである。従来のグレースケール画像処理のモルフォロジー処理法を単純にカラー画像のへ各チャンネルに拡張するやり方はよくない。

構造要素内の画素間の大小関係を決めるために、各チャンネルに優先度を付けるなど、多数の提案がなされてきた^{[10][10][12]}。その中の基本的な考え方の一つは、計算・用意された重みベクトルによって、各チャンネルの画素値を調整し、その結果を用いて画素間の大小関係を作ることである。しかし、その重みベクトルがどのように設定するか、まだその設定が人間の視覚特徴に合致するかどうかなどの疑問はかなり残っている。

画素値ベクトルのノルムをユークリッド距離で定義し、その大きさを使って画素間の大小関係を決めるやり方もよく使われている^[13]。しかし同じノルムの画素間の大小関係を決めることが難しい。例えば、赤色(255, 0, 0)と緑色(0, 255, 0)は同じで大きさのノルムを持っているため、赤色と緑色のどちらが大きいかは事前に決めると汎用性がなくなる。本研究では、画像の内容に応じて動的に画素間の大小関係を画素の顕著度で定義し直すと、二値画像にも、多値画像にもカラー画像にも矛盾のない一般的なモルフォロジー処理ができる。

3.3 顕著度マップによるモルフォロジー処理

本節は画像の顕著度マップと画素の「目立つ度合」を基準として、モルフォロジー処理に必要な画素間の大小関係を定義する。

図 3.3 は顕著度マップによる収縮処理(erosion)の計算過程を説明するために作られた画像である。図 3.3 (A)は 5×5 の人工画像を対象とする。画像中の画素はすべて彩度 1.0 に、明度 1.0 に設定する。色相の値は 3.3 (A)に記した数値とする。この収縮過程は中心画素 $f(2,2)$ (中心画素)のみを対象とし、3×3 の正方形構造要素で処理を行う。

最初は、構造要素内の顕著度マップ $\overline{c_r(x,y)}$ を算出し、その結果は図 3.3(B)に書かれている。一般的に、構造要素内の各画素自身の顕著度は注目画素と違う構造要素で計算されている。そのため、注目画素の構造要素だけではなく、その範囲外の画素も間接的に構造要素内の顕著度分布に影響を与える。n×n(n は奇数)の正方形構造において、外側の画素の構造を考慮して、注目画素は最大 $(2n-1) \times (2n-1)$ 範囲までの画素の情報を利用することになる。顕著度マップのベクトル値に対応する画素の大小関係を定義するために、式 2.5 で計算された正規化顕著度のノルムを利用する。図 3.3(C)は式 2.6 から式 2.5 に逆計算した構造要素内の顕著度 C_s である。

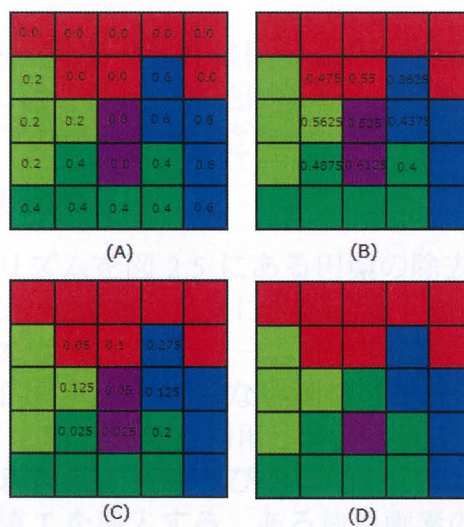


図 3.3 モルフォロジー処理例

fig. 3.3 An example of morphological process for color image

算出されたデータを用いて画素間の大小順序を作る。その大小順序を用いて、式 3.3 で注目画素の構造内にある正規化された顕著度 C_s が最も小さな画素の位置を探し出す。構造要素内で最小顕著度の画素が複数個ある場合、注目画素に近い方の画素の位置を利用する。算出された結果は変数 (\hat{x}, \hat{y}) に記録する。

$$(\hat{x}, \hat{y}) = \arg \min_{(x, y)} \left\{ \begin{array}{l} \overline{\|c_s(x-n, y-n)\|} \\ \dots \\ \overline{\|c_s(x+n, y+n)\|} \end{array} \right\} \quad (3.3)$$

図 3.3(C)は式 3.3 を使って求めた注目画素の構造範囲内の各画素の正規化顕著度 C_s のノルムである。

膨張処理の場合、注目画素の構造内にある正規化された顕著度 C_s のノルムが最大の画素の位置は式 3.4 で計算する。

$$(\hat{x}, \hat{y}) = \arg \max_{(x, y)} \left\{ \begin{array}{l} \overline{\|c_s(x-n, y-n)\|} \\ \dots \\ \overline{\|c_s(x+n, y+n)\|} \end{array} \right\} \quad (3.4)$$

最後、式 3.5 で注目画素の値を位置 (\hat{x}, \hat{y}) の画素の値で置き換える。

$$\overline{f_{new}(x, y)} = \overline{f_{origin}(\hat{x}, \hat{y})} \quad (3.5)$$

収縮処理の場合、注目画素 $\overline{f(2,2)}$ は図 3.3(C)の構造要素内の最小値である $\overline{f(1,3)}$ (0.025)の画素値に置き換え、緑色になる。その結果は、 $\overline{f(2,2)}$ の 3×3 構造要素内において、左下の緑が最も目立たない色を意味する。

3.4 実験及び従来法との比較

実験 1 は提案アルゴリズムを図 2.5 にある円環の除去に適用したものである。図 3.4(A)が示すように、元画像の中央にある円環と背景のカラー円との色関係は複雑なものである。図 3.4(A)のような一方的に背景の色が数値的に大きい、もしくは円環の色が数値的に大きいではない画像の処理は、従来法で処理しづらい。提案手法画と 5×5 の構造要素を利用し、顕著度マップによるモルフォロジー収縮処理を行う。処理時間の短縮及び処理能力の柔軟性を考え、顕著度のノルムに適応するしきい値 T を導入する。ある注目画素の C_s のノルムがしきい値 T 以下の場合、その処理過程は省略される。画像の顕著さが低い画素を処理対象から外せることで、そのような領域のデータの劣化避けるほかに、処理の計算

量を減らすこともできる。しきい値 T を 0.300 に設定し、まず図 3.4 (B) のような中間結果が得る。図 3.4(B)において、太い円環が大分背景の色に変わったが、周りの色分布によって一部はノイズとして残っている。同じパラメータで図 3.4 (B) の中間結果に対してもう一度同じ処理過程を施すと、図 3.4 (C) に示すような結果画像になり、色と形状が鮮明であった円環は完全に消された。それ以外の部分の画素は処理範囲外であるため、まったく影響を受けずに元画像への画質劣化はほとんど発生しなかった。

実験 2 は、ホワイトノイズを除去する実験である。図 3.5 (A) はホワイトノイズを加えた lenna 画像である。しきい値 T を 0.20 に設定し、顕著度マップによるモルフォロジー収縮処理を一回施した結果、図 3.5 (B) のように、画質の影響を与えずにノイズの大部が除去した。帽子の右側や肩だけ、ノイズと周りの画素値が似ているため、処理効果はよくない。さらにしきい値を 0.09 までに調整し、もう一度アルゴリズムを施すと、ノイズは完全に除去され、処理対象は図 3.5 (C) になる。やや不満なのは、図 3.5 (C) は低いしきい値を使ったため、ノイズ以外の画素への影響は高い。

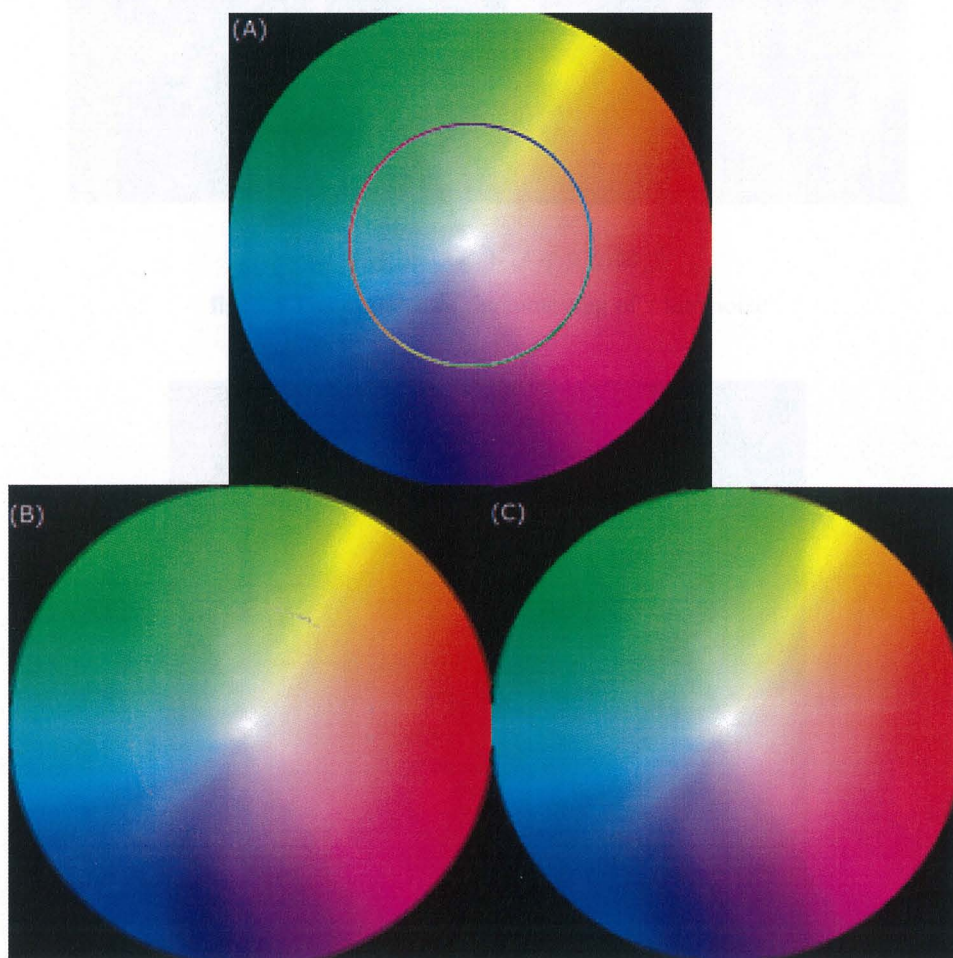
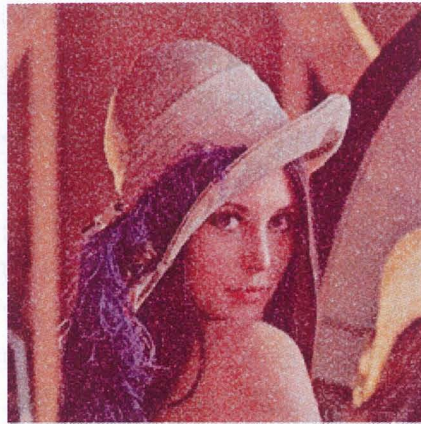


図 3.4 円環除去

fig. 3.4 The result of the removing of circular ring



(A)



(B)

(C)

図 3.5 ホワイトノイズの除去
fig. 3.5 The example of the removing of white noise



図 3.6 カラーノイズ画像
fig. 3.6 Color image with complicated noise

実験3は、カラー画像の小ブロックや線状ノイズの除去実験である。図3.6のノイズはペイントツールを使って元画像に適当に加えたものである。特に右の黄色の塊はノイズとしての大きさが普通程度以上であり、周りの色関係も光の照射により複雑になっている。このような画像は従来法で処理しづらいということも明白である。

提案手法を用いて、しきい値0.164に設定し、元画像にアルゴリズムを施すと、図3.7(A)の元画像は図3.7(B)に変化した。その図が示すように、メインの黄色の塊以外のノイズは大部消されたか、薄くなっているかである。続いて、しきい値0.080で処理をもう一度行い、図3.7にある赤の円の中の黄色の塊状のノイズの大きさを縮めさせることができた。さらに、しきい値を0.60までに上げ、繰り返して処理すると、図3.7(D)～図3.7(G)のように塊状のノイズの大きさは減る一方、最後完全に消えてなくなった。右の方は左の画像の赤い円で囲まれた部分の拡大表示である。この拡大図から、元来の緑色の上に覆われていた黄色の塊がなくなり、それは周囲のオレンジの緑色にある程度回復した。この処理で、近くのハイライトを表現する画素への悪影響は小さく抑えられている。

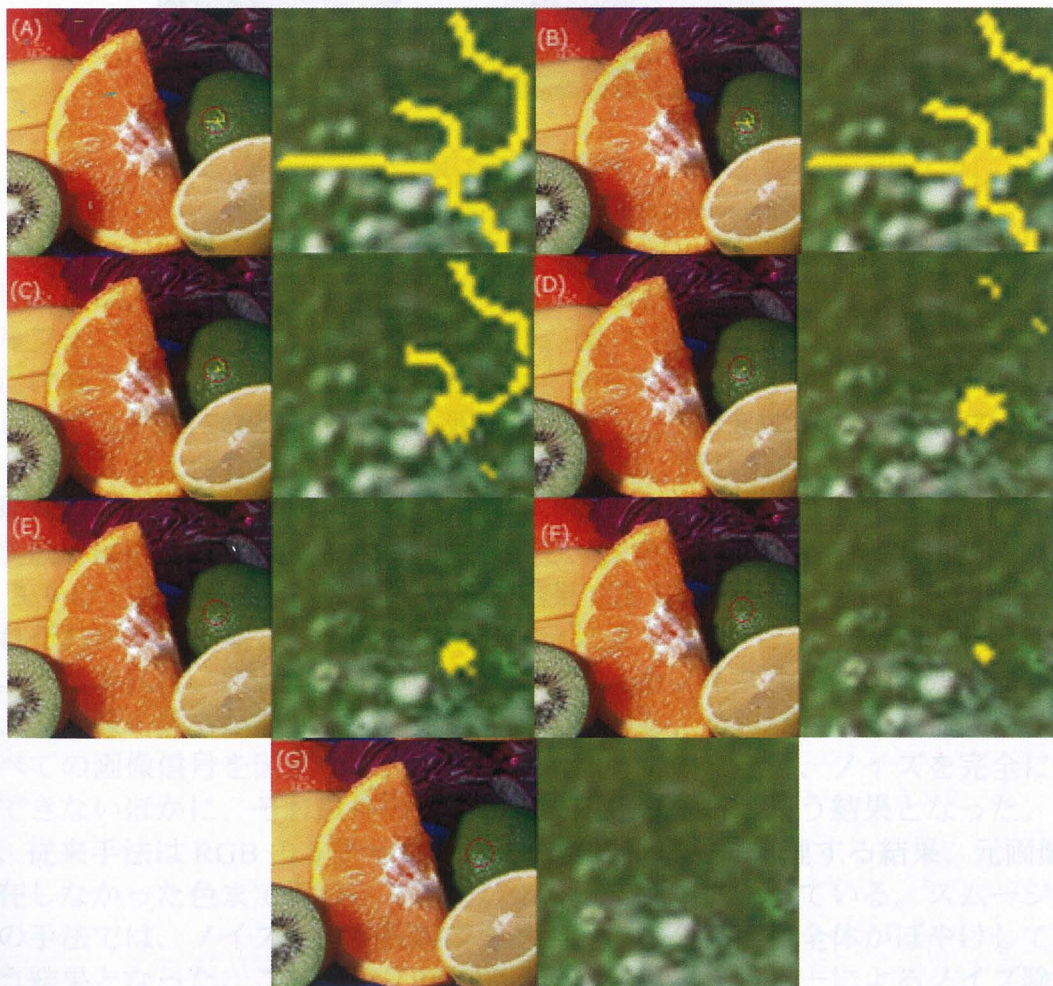


図 3.7 カラーノイズの除去

fig. 3.7 The removing of complicated color noise

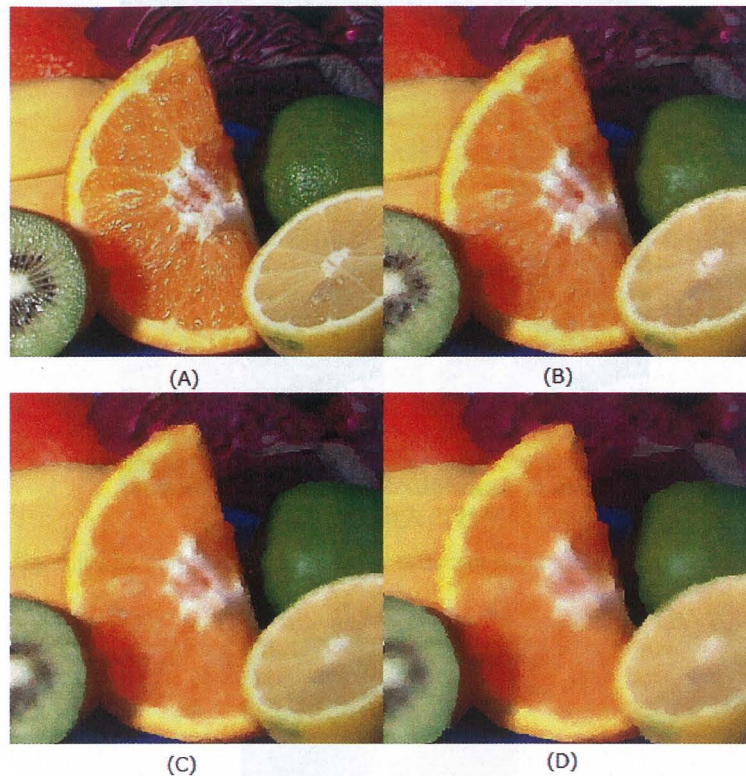


図 3.8 収縮処理

fig. 3.8 The erosion process for an ordinary color image

実験 4 は、しきい値 T をゼロに設定し、全画素に対して顕著度マップによるモルフォロジー処理を行ったものである。図 3.8(A)の元画像に、提案手法を一回だけ施すと、処理対象は図 3.8(B)のようなグランドグラス (ground glass) を通して見る効果になる。されに一回実行すると、図 3.8 (C)が示すように、グランドグラスのようなぼやけはさらに深刻となる。同じ処理を 10 回繰り返すと、処理対象の変化はだんだん小さくなり、最後図 3.8(D)のように収束した。

実験 5 は図 3.6 の画像を処理対象として、従来法と提案手法との比較である。図 3.9 の上部の画像は提案手法の処理結果であり、下部の左側と右側の画像はそれぞれ従来の収縮処理と線形フィルタによるスムージング処理の結果である。提案手法は、顕著度によって人為的に加えたノイズを集中的に処理する一方、元画像の緩やかな部分への影響を抑えている。それに対して、従来手法はすべての画像信号を区別なく、まとめて処理しているため、ノイズを完全に除去できないほかに、それ以外の信号までを劣化させてしまう結果となった。また、従来手法は RGB の 3 チャンネルにそれぞれ単独で処理する結果、元画像に存在しなかった色まで現れ、ノイズがかえって顕著になっている。スムージングの手法では、ノイズを少し軽減できるが、その分、画像全体がぼやけしてしまう結果となった。この画像に対してメディアンフィルタによるノイズ除去も試したが、人為的に加えた線状ノイズやカラーの小ブロックはほとんど除去できなかった。

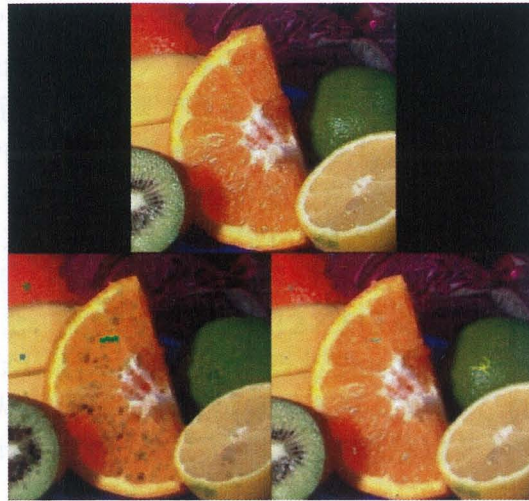


図 3.9 ノイズ除去の効果の比較

fig. 3.9 The comparison results for different noise removing methods

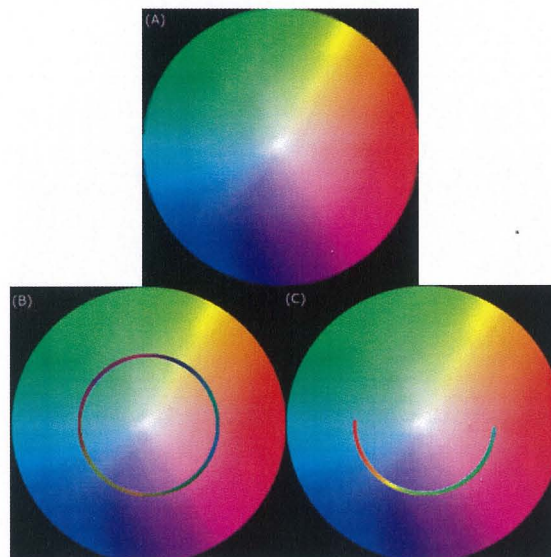


図 3.10 円環除去の効果の比較

fig. 3.10 The comparison results of the removing of circular ring

図 3.10 は実験 1 の画像を処理対象とし、提案手法と従来法による処理結果の比較である。図 3.10 (A)は提案手法の処理結果であり、図 3.10 (B)は従来のグレースケール画像の収縮処理を R、G、B チャンネルでそれぞれ単独適用した結果である。図 3.10 (C)は元画像を HSV 色空間に変換してから、H、S、V チャンネルにそれぞれ従来の収縮処理を施したものである。図 3.10 (B)は円環を軽減するのではなく、かえって顕著させてしまった。図 3.10 (C)の結果は片方が軽減できたが、もう片方は逆に太くなった。以上の対比実験から、提案手法は元画像の信号を最大限に保存しながらノイズ除去することを達成している。

提案手法は Windows OS の環境で Microsoft 社製 Visual Studio 2008 を使って実

装した。コンピュータの CPU は Intel Core i7 940 2.93GHz で、内臓メモリは 4 GB、まだ ATI Radeon HD4870 のグラフィックカードを使用している。提案手法は、各々の画素において独立な処理できることと中間結果の顕著度マップを用意することによって、GPU による高速処理で実現している。計算時間を比較するために、CPU 処理だけを利用して提案手法を別途でプログラムを作成した。256×240 の画像サイズにおいて、CPU プログラムの方は、47ms の処理時間が必要であった。これに対し、GPU を使った処理では、1000 回繰り返して得た平均処理時間は 4ms であった。画像サイズを 512×512 の場合、CPU だけのプログラムは 188ms がかったが、GPU で処理すると 5ms であった。さらに 1024×721 と 1758×988 までに拡大した場合も、画像サイズに応じて CPU のアルゴリズムは線形的に処理時間が増加したが、GPU による処理はそれぞれ 7ms と 12ms でできた。

4. 顕著度マップによる補間法

4.1 背景

近年、電子技術の進捗と共に、カメラやビデオカメラに代表とされる撮影機器の解像度が大幅に上がってきた。同時に、ディスプレイやプリンターに代表される表示機器の解像度も改善してきている。しかし表示機器の解像度と撮影機器の解像度が一致していない場合は多くあるため、撮影機器より得られた元画像のデータから表示機器の解像度に合わせて画像の拡大縮小処理をする必要がある。デジタル画像は画素によって離散的に表現されているため、拡大や縮小による画素数の変動は、画素を割り増ししたり、間引いたりすることが発生する。それによって画質の劣化が避けられない。

画像の拡大縮小処理の目標は元画像の特徴を保留する上に、より自然と鮮明な結果画像を作ることである。拡大縮小処理を行う際に、目標画素が元画像に対応する場所をベースにして、一定範囲内の元画像の画素をサンプリングする必要がある。そのサンプリングの範囲は画像の拡大縮小処理のアルゴリズムを大きく二つの種類に分ける。それぞれはスケーリング倍率が小さい処理過程に適応する固定サンプリング範囲のアルゴリズムとスケーリング倍率が高い処理過程に適応する動的なサンプリング範囲のアルゴリズムである。ほかに、画像の内容によって、自然画像に適応するアルゴリズムと人工画像に適応するアルゴリズムなどの分類方法もある。また、画像のスケーリングに限らず、画像の補間法は画像のコーディング(coding)、調整、合成、圧縮、転送、顔認識など多数の研究課題にも重要な役割を果たしている。

4.2 節では従来の画像のスケーリング処理法を紹介する。4.3 節では提案手法の重要なパラメータ”面積 $S(i, j)$ ”の計算方法を説明するし、4.4 節でそれを使い、画像の補間方法を述べる。4.5 節では、比較実験を持って提案アルゴリズムの有効性を示す。

4.2 従来の補間法

画像のスケーリングを行うには、まずスケーリング比率を利用して、目標画像の注目画素の位置に対応する元画像の位置を計算する。その過程は式 4.1 で示す。

$$(x_o, y_o) = (x_n, y_n) \times \frac{1}{Ratio} \quad (4.1)$$

(x_n, y_n) は目標画像の注目画素であり、 (x_o, y_o) は注目画素が元画像に対応位置である。ratio は元画像から、新画像へのスケーリング比率である。一般的に、注目画素の位置 (x_n, y_n) は整数であるが、対応画素 (x_o, y_o) の位置は浮動小数になる可能性が高い。浮動小数で示す座標における画素値を周りの整数座標しか持た

ない画素より算出するために、補間を使う必要がある。

画像のスケーリング処理に使う補間法の中、最もシンプルなもの最近傍法 (Nearest Neighbor) である。それはまず式 4.1 を使って注目画素に対応する元画像の位置 (x_0, y_0) を計算する。そして元画像から、 (x_0, y_0) に最も近い画素を取り出し、その画素値を注目画素の値に代入する。最近傍法のアルゴリズムは簡単であり、計算量も少ない。この方法は近似であるため元画像の特徴を失って、エイリアシング (aliasing) の現象が高い確率で発生する。

双線形補間法^[14] (Bilinear) は、注目画素が元画像に対応する位置の近傍 4 点の画素までの距離とその 4 つの画素の値を重き付き平均計算によって求められる。この方法は複数の画素をサンプリングによって、目標画素のエイリアシング発生をある程度抑える。しかし、複数のサンプリング画素の情報を一つの目標画素に集約することで、画像全体がぼやけてしまう。最近傍法と比べ、双線型法によって計算された画像の質はよい。

双三次 cubic 法はさらにサンプリング範囲を拡大し、近傍 4×4 の画素の値を利用する。それらの画素をベースに補間する計算式によって、双三次 cubic 法は B-Spline 法^{[15][16]}、たたみ込み内挿法^[17] など数種類に細分できる。cubic 法もマルチサンプリングから画素値を求めるため、双線型法と同じようにぼやけが発生することがあるが、双線型法より画質が改善されている。一つの画素を計算するのに、16 個の非隣接メモリから画素値を読み取り、複雑な関数計算を行う双三次法はアルゴリズムの計算量の増加により、実時間の反応が要求されるシステムへの応用が制限されている。

面積平均法のサンプリング範囲は画像のスケーリング倍率により動的に決められる。固定的なサンプリング範囲を用いる画像スケーリングアルゴリズムにおいて、縮小倍率が高くなってくると、必ず結果画像に影響を与えないが画素が出てくる。その結果、画像を間引く処理の場合、情報の欠落は避けられないため、画像全体の円滑さが失ってしまう。それに対して、面積平均法は動的に設定するサンプリング範囲は必ず画像全体を覆うため、情報の欠落はなく高倍率の画像縮小過程に高品質の結果を得られる。その反面、サンプリング範囲に覆われている画素を全部同一に処理され、顕著になっている画素は薄められてしまう。

従来は画像内のエッジ^[18] と輪郭の特徴、フリエ変換やウェレット変換後の特徴などを利用する補間法が多数研究されている。本章は顕著度マップを利用してスケーリング結果の円滑性と元画像の中の顕著な特性を保持するトレードオフを考えて新しい面積積分法による補間方法を提案する。

4.3 画素のローカル投影面積

スケーリング処理で目標画素の画素値を求める時、従来の補間法では、目標画素が対応する元画像上の点を中心にある近傍の画素を利用して様々な補間の計算式で計算をしている。それらの手法は境界問題を考慮する必要がある。例えば、目標画像の画素は元画像 $(0, 0)$ 画素に対応した場合、 $(0, 0)$ 画素を中心とする近傍画素の一部は元画像の外側にあり、実質上存在しない画素を考慮に入れ

ることになる。本論文では、違う方法で目標画素のサンプリング範囲を決めて、従来法のような境界問題が発生しない。図 4.1 に示されるように、提案手法は目標画像の解像度に応じて元画像をバーチャルな格子で分割する。破線が囲んだ格子は目標画像の一つの画素のサンプリング範囲となる。以下は、そのバーチャルな格子を利用して、目標画素の値をバーチャルの格子が占める元画像の面積とバーチャルの格子における画素の顕著度を統合して求めるようにする。

任意の注目画素のサンプリング範囲の初期点を $P_b(x_o, y_o)$ で表記する。注目画素に対応するバーチャル格子は P_b から始め、スケーリング倍率で決めた長さの矩形である。その矩形の終点 P_e を式 4.2 で計算する。

$$P_e = (x_o + \frac{1}{ratio.x}, y_o + \frac{1}{ratio.y}) \quad (4.2)$$

スケーリング処理は拡大と縮小の二通りの処理がある。縮小処理の場合は目標画像の1つの画素は1つ以上の元画像の画素に対応する。

元画像の画素サイズを 1×1 とする。目標画素のサンプリング範囲内のすべての画素がバーチャル格子に占められている面積をローカル投影面積 $S(i, j)$ とする。ただし、 P_b を含んでいる画素の相対位置を $(0, 0)$ として、 (i, j) はバーチャルの格子にある画素の相対位置である。

ローカル投影面積を計算する前に、まず図 4.2 に示されるようなバーチャル格子の虚線からその中の実線までの距離 lx, rx, ty, by を求める必要がある。

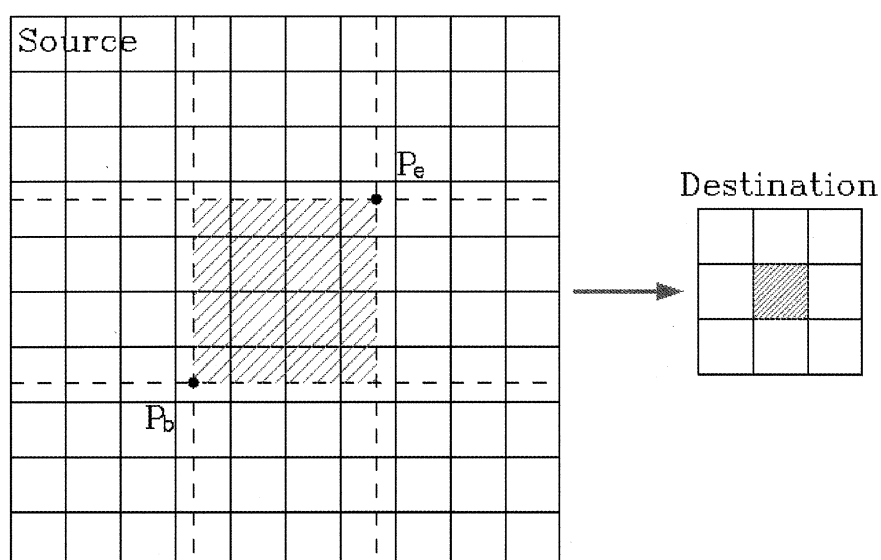


図 4.1 目標画像による原画像の領域分割

Fig. 4.1 The segmentation of source image based on the size of destination image

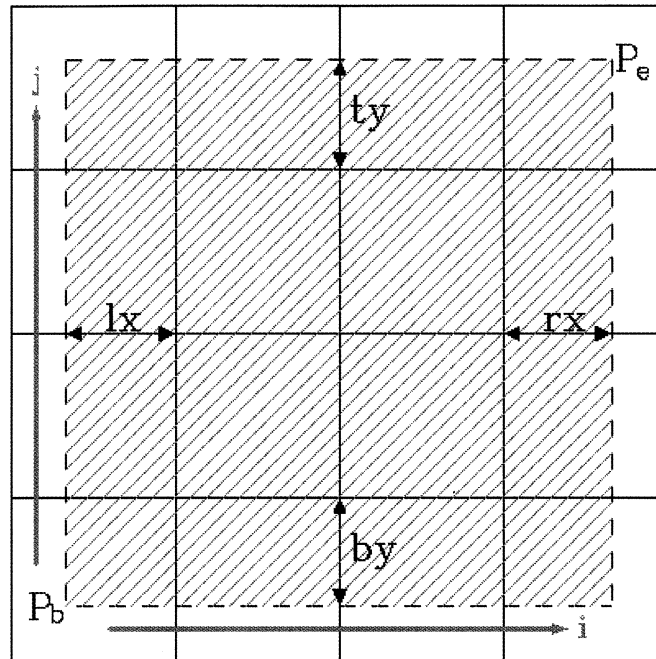


図 4.2 投影範囲

Fig 4.2 The projection range

$$\begin{cases} lx = 1 - \text{fract}(P_b.x) \\ rx = \text{fract}(P_e.x) \\ by = 1 - \text{fract}(P_b.y) \\ ty = \text{fract}(P_e.y) \end{cases} \quad (4.3)$$

$$\begin{cases} (i, j)_{\min} = (0, 0) \\ (i, j)_{\max} = \text{ceil}(P_e) - \text{floor}(P_b) - 1 \end{cases} \quad (4.4)$$

式 4.3 と式 4.4 において $\text{fract}(\cdot)$ は変数の小数部分を求める関数である。 $\text{ceil}(\cdot)$ は \cdot より大きい最小の整数を求める関数である。 $\text{floor}(\cdot)$ は \cdot より小さい最大の整数を求める関数である。

拡大の場合、バーチャル格子の始点 P_b と終点 P_e は元画像の同じ画素に存在することが多い。拡張倍率によって、バーチャル格子は図 4.3 に示される可能性がある。バーチャル格子は元画像の一つの画素の中にある場合と二つの元画像の画素に跨っている場合と 4 つの元画素に跨っている場合はあるが、図 4.3 の (A)(B)(C) の図に示される 3 つの場合に分解することができる。拡大処理においては、式 4.3 で計算された境界パラメータ lx, rx, by, ty などは式 4.5 で計算する。

$$\begin{cases} lx = rx = P_e \cdot x - P_b \cdot x, & \text{if } i_{\max} = i_{\min} \\ by = ty = P_e \cdot y - P_b \cdot y, & \text{if } j_{\max} = j_{\min} \end{cases} \quad (4.5)$$

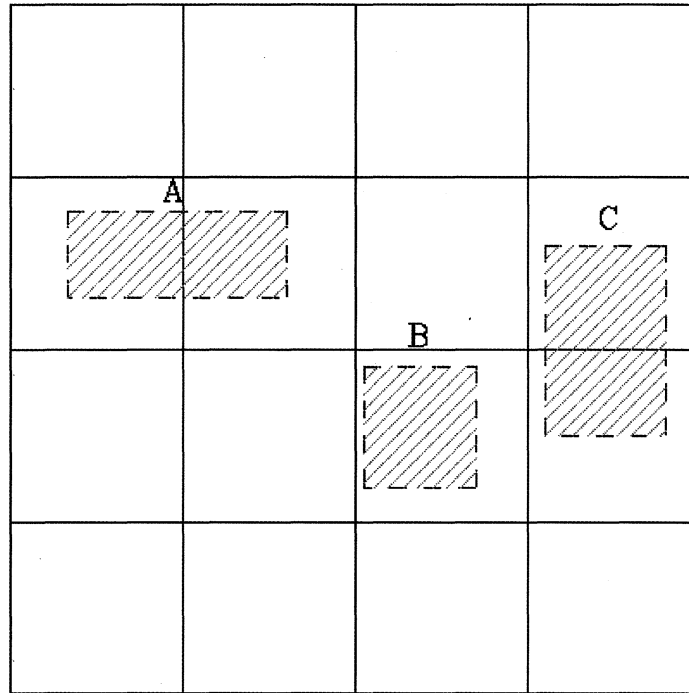


図 4.3 拡大処理による可能な投影

Fig. 4.3 Projection in image enlargement

$$S(i, j) = \begin{cases} lx \times by, & i = i_{\min} \ \& \ j = j_{\min} \\ rx \times by, & i = i_{\max} \ \& \ j = j_{\min} \\ rx \times ty, & i = i_{\max} \ \& \ j = j_{\max} \\ lx \times ty, & i = i_{\min} \ \& \ j = j_{\max} \\ lx \times 1, & i = i_{\min} \ \& \ j_{\min} < j < j_{\max} \\ rx \times 1, & i = i_{\max} \ \& \ j_{\min} < j < j_{\max} \\ by \times 1, & j = j_{\min} \ \& \ i_{\min} < i < i_{\max} \\ ty \times 1, & j = j_{\max} \ \& \ i_{\min} < i < i_{\max} \\ 1 \times 1, & \text{others} \end{cases} \quad (4.6)$$

lx, rx, ty, by のパラメータが求めたら、バーチャル格子中の影部分の画素の面積 $S(i, j)$ は式 4.6 で求められる。

4.4 顕著度マップを利用した面積平均法

バーチャル格子から対応する注目画素の画素値を求める際に、前節で求めたローカル投影面積を重み係数として利用できるが、バーチャル格子内のすべての画素は同一視されてしまって、それぞれが目標画素値への貢献度はローカル投影面積だけによってしまう。このやり方は境界条件の判定方法を除けば従来の面積平均法と本質的に一緒となる。本研究では、特に高い倍率の画像縮小において、縮小によって元画像の詳細をできるだけ温存させるために、バーチャル格子内のすべての画素の顕著度を加味した新しい重み因子 $C_t(x, y, i, j)$ を導入する。この画素重み因子は式 4.7 で定義する。

$$C_t(x, y, i, j) = \overline{C_s(x_o, y_o)} \times S(i, j) + 1.5 \quad (4.7)$$

顕著度 $\overline{C_s(x, y)}$ のそれぞれの要素の値の範囲はそれぞれ $[-0.5, 0.5]$ 、 $[-1.0, 1.0]$ 、 $[-1.0, 1.0]$ であるため、そのノルムの最大値は 1.5 となるため、式 4.7 において正規化処理をしてその重み因子の値の範囲を $[0.0, 1.0]$ にした。

この重み因子を利用して、画像の拡大処理にも縮小処理にも同一原理で行うことができる。この方法を使えば従来のスケーリング処理アルゴリズムと同等あるいはそれ以上の画質効果を得ることができた。

画像の縮小処理において、縮小倍率が高ければ、結果画像の 1 つ 1 つの画素値はより多くの元画像の画素値に影響される。もしすべての画素を同じように面積だけでその貢献しか計らなければ、画像の重要な情報が縮小することによって失ってしまう。例えば、白い大きい領域の中に 1 つの黒点があるような画像を考えて見る。画像をどんどん縮小していく過程で黒い点が薄れていき、最後に完全に消えてしまうことになる。本研究ではスケーリングの倍率を考慮に入れて重み因子をさらに調整して式 4.7 を式 4.8 のように改善した。

$$\begin{cases} C_p(n, x, y, i, j) = C_t(x, y, i, j)^n \\ n = \frac{1}{ratio} \end{cases} \quad (4.8)$$

画像の縮小処理において、縮小倍率が高いほど目標画素が対応するバーチャル格子が大きくなるので、その中の顕著な画素の性質を残すために、指数 n を導入した。 $0 < C_t < 1.0$ であるため、 n が大きくなるにつれて、重み因子 C_p は重み因子 C_t よりも顕著度を強調できる。つまり、ゼロより大きい 1 より小さい数を n の指数計算をすることによって、小さい小数は大きい小数より小さくなっていくスピードが大きくなる。よって、画像を縮小していく過程においては、顕著度が小さい画素の影響が先になくなり、顕著度がより高い画素の影響はより保持できるような効果が得られる。

重み因子 C_p を用いて、縮小画像の結果画素の値 $\overline{Tex}_n(x,y)$ を式 4.9 で計算する。

$$\overline{Tex}_n(x,y) = \frac{\sum_{i=0, j=0}^{i=i_{\max}, j=j_{\max}} \left[\overline{Tex}_o(x_o, y_o) \times C_p \left(\frac{1.0}{ratio}, x_o, y_o, i, j \right) \right]}{\sum_{i=0, j=0}^{i=i_{\max}, j=j_{\max}} C_p \left(\frac{1.0}{ratio}, x_o, y_o, i, j \right)} \quad (4.9)$$

$\overline{Tex}_n(x,y)$ は (x,y) における結果画像の画素値であり、 $\overline{Tex}_o(x,y)$ は (x,y) における元画像の画素値である。

4.5 実験及び従来法との比較

実験 1 は図 4.4(A) に示されている風景画像を 16%(281×158) までに縮小する処理を従来法と提案手法で比較するものである。元画像図 4.4(A) の中、天空、草と川の漣と三つの部分の画素値の変化は違う様子となっている。画像上部の青空の部分の画素値の変化は緩やかである。手前にある草の重ね合いと日陰の影響で画素値の変化は天空領域より相対的に激しくなっている。画像の左下の葉と川の部分に光の反射で生成されたハイライトがいくつも存在する。この領域の画素値の変化は最も激しい。この画像を従来法と提案手法で処理した結果は以下のものである。図 4.4(B) は最近傍法の結果であり、図 4.4(C) は双線型法の結果である。図 4.4(D) は双三次法(B-Spline) で、図 4.4(E) は面積平均法で、図 4.4(F) (5×5 構造) は顕著度による面積平均法での処理結果である。

画素値の変化が緩やかである元画像の青空部分を観察する。それぞれの処理結果と元画像を比べて、青色の変化と分布はほぼ同じであり、不自然さは見当たらない。全のアルゴリズムはその部分に対して良い処理結果が得られている。

画像の明暗変化などを含む元画像の草の部分の処理結果を観察する。図 4.5(A)~図 4.5(F) は図 4.4(A)~図 4.4(F) の草の部分の拡大図である。それぞれの図は順番で元画像と各アルゴリズムの結果画像と対応している。図 4.5(B)、(C)、(D) の処理結果と図 4.4(E)、(F) の処理結果を比べて見ると、図 4.5(B)、(C)、(D) の結果の方は画面がより滑らかである。その原因は、図 4.6 に示されているように、高倍率の画像縮小過程において、目標画像の隣接画素が元画像におけるサンプリング範囲の間に、サンプリングされていない画素が存在するからである。例えば、図 4.6 の 12×12 の元画像を 2×2 の目標画像に変換する。図 4.6 右の目標画像の ABCD 画素が左の元画像の ABCD 画素と対応している。従来法の中のサンプリング範囲が大きな双三次法(4×4) のを用いて、左の点 A のサンプリング範囲は他の点のサンプリング範囲と離れている。そして、点 A のサンプリング範囲と他の点サンプリング範囲との関連性を失ってしまう。よって、どんな方法で目標画像の画素 ABCD を計算しても、その間の連続性を保つ保証はできない。他に静的なサンプリング範囲を利用するアルゴリズムは必ずこのような問題が存在する。これはいわゆる「間引き」問題である。一方、動的にサンプリング範囲を有する面積平均法と顕著度による面積平均法のサンプリング範囲は

スケーリング倍率によって変わるため、元画像の全体を覆う保証ができる。それによって得られた結果画像図 4.5(E)(F)の円滑さは保たれている。

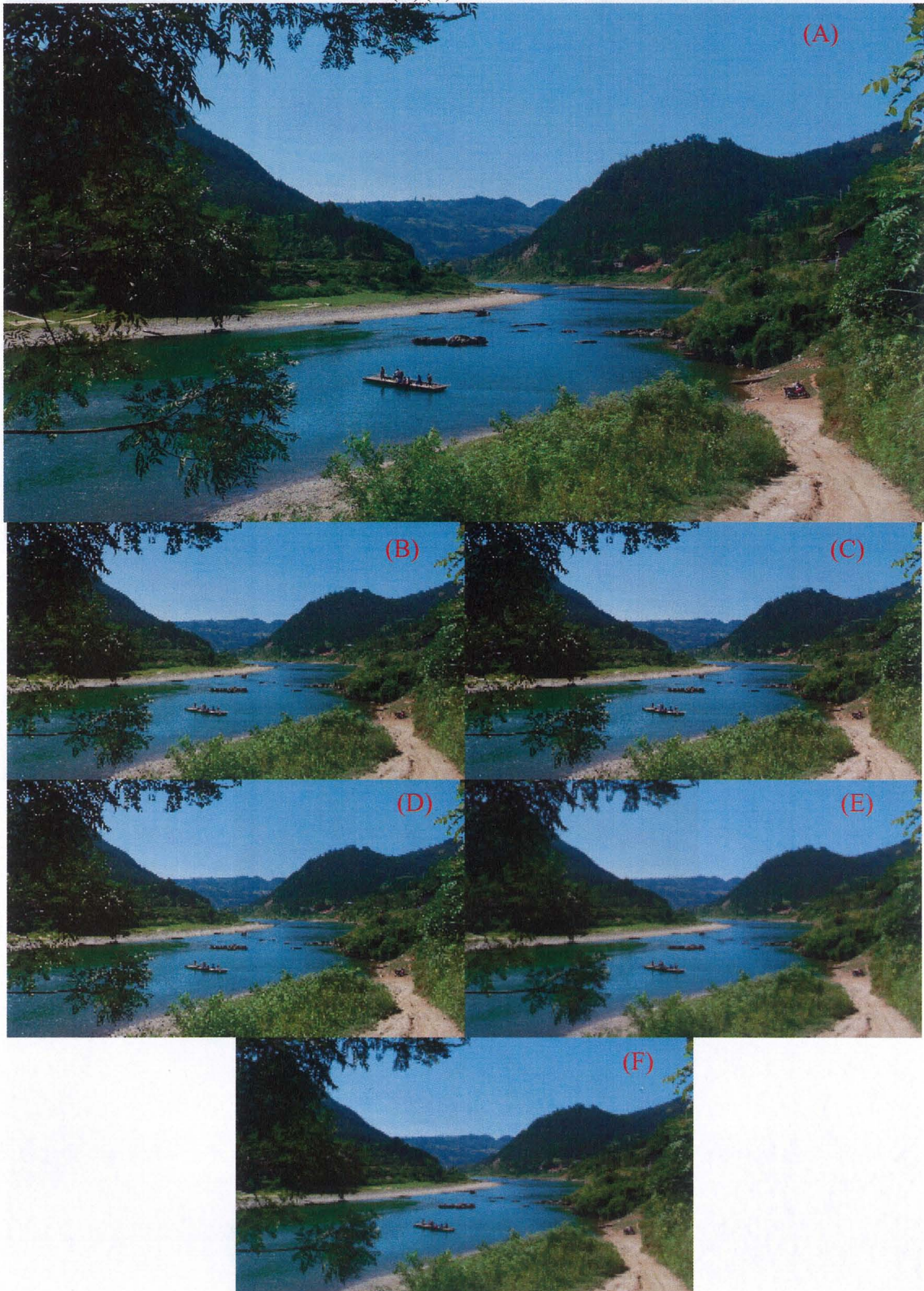


図 4.4 風景画像による縮小処理
Fig. 4.4 The reduction process on natural image



図 4.5 草の部分の拡大図
 Fig. 4.5 The zoom in image of the grass

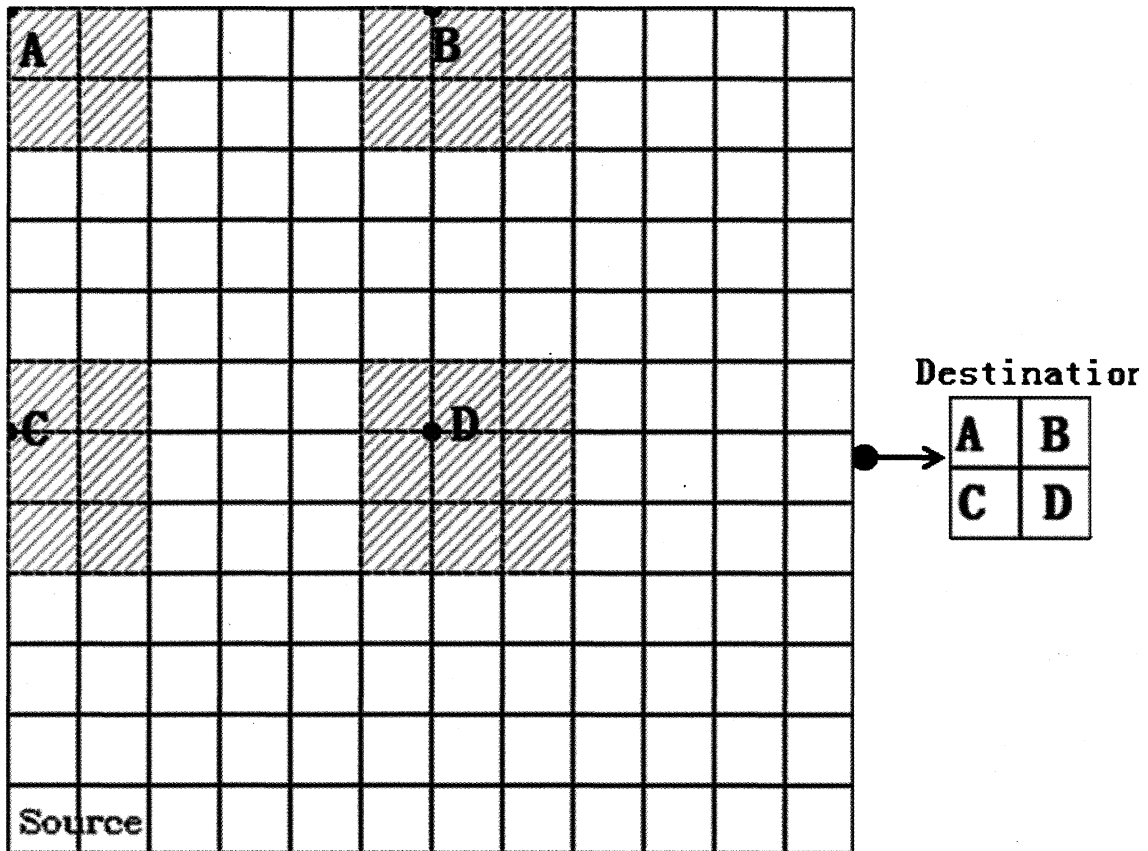


図 4.6 投影範囲の間の空白領域

Fig. 4.6 The space between projection

ハイライトを複数含む元画像の葉と川の部分の処理結果を観察する。図 4.7(A)~図 4.7(F)は図 4.4(A)~図 4.4(F)の葉と川の部分の拡大図である。それぞれの図は順番で元画像と各アルゴリズムの処理結果画像と対応している。図 4.5 と同じように、静的なサンプリング範囲の処理法で得られた結果画像図 4.7(B)、(C)、(D)の結果には不自然さが生じている。それに対し、図 4.7(E)、(F)の結果画像は前述した草部分の画像と同様にその滑らかさが保たれている。しかし、従来の面積平均法の結果画像である図 4.7(E)において、図 4.7(A)の中に赤い円で囲まれたハイライトは見えなほど薄くなっていた。さらに縮小倍率を上げると、目標画素の値を計算する加重平均に参加する画素は増えるため、このハイライトはどんどん薄くなり、最後に消えてしまう。それに対して、画像の顕著度を考慮に入れた提案手法は画像の円滑さを保った上、画像の特徴であるハイライトをある程度保持した。縮小倍率を高くなっても、縮小倍率に応じて顕著度の影響をその画素の重みで反映すれば、元画像の顕著なところの情報は消えないようにコントロールできる。

図 4.7 樹の部分の拡大図

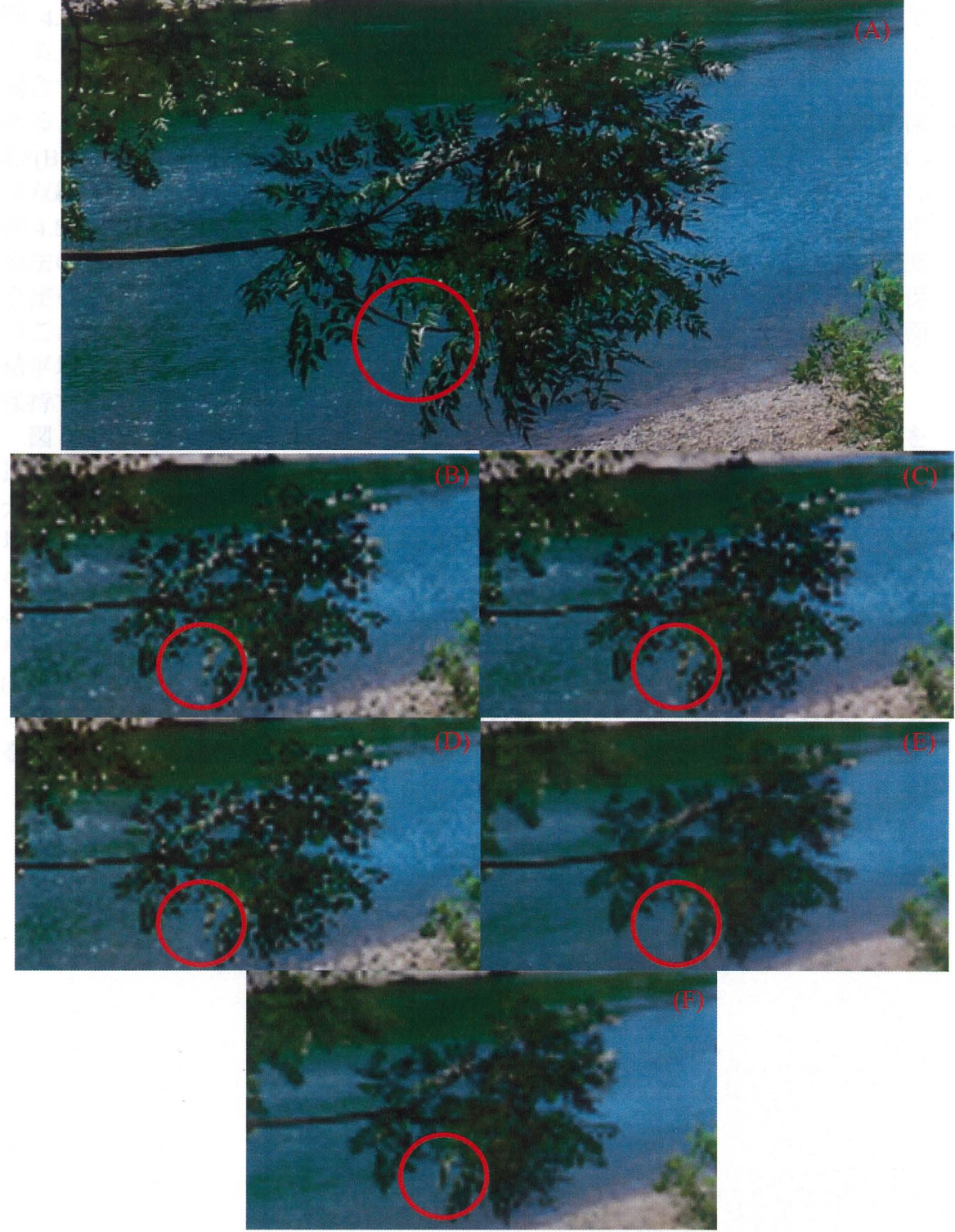
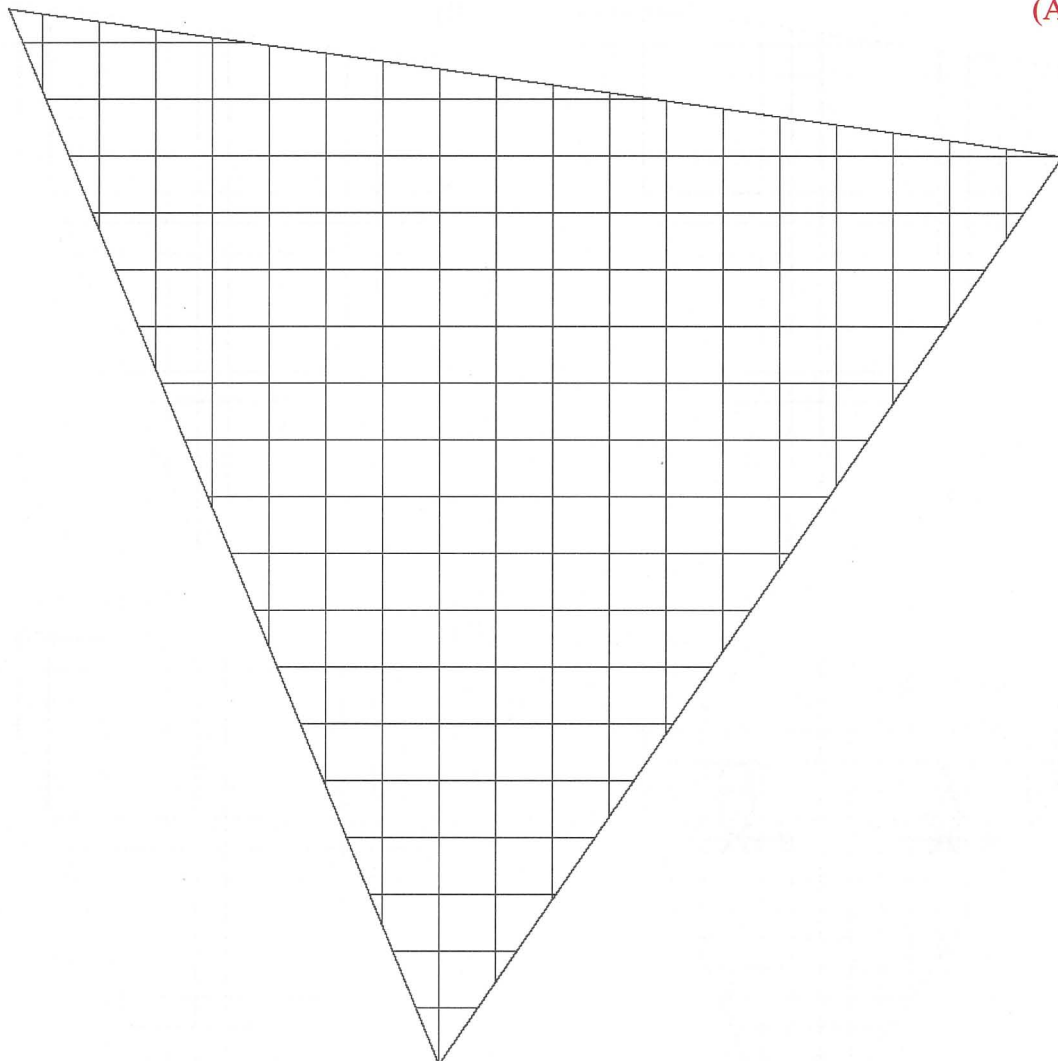


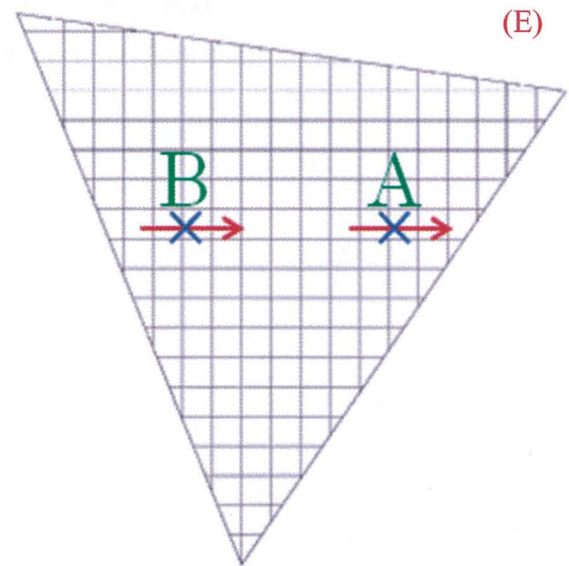
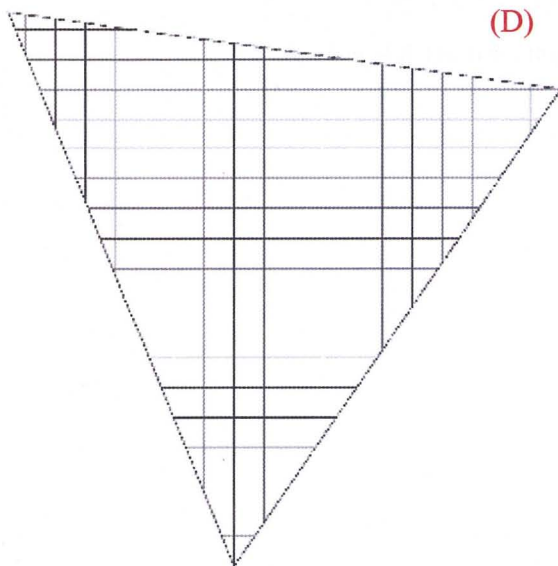
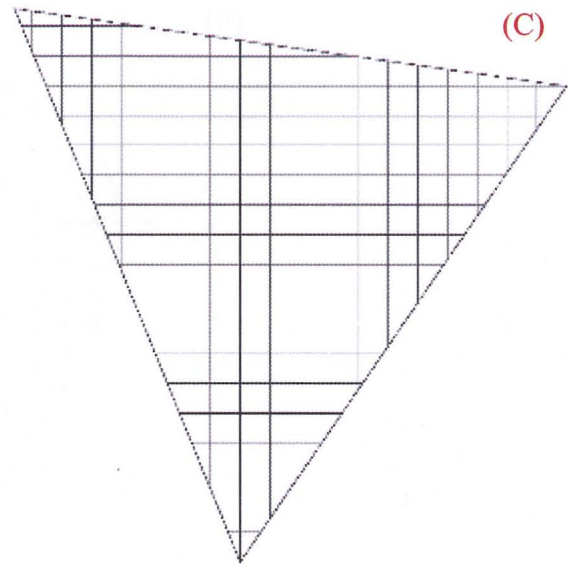
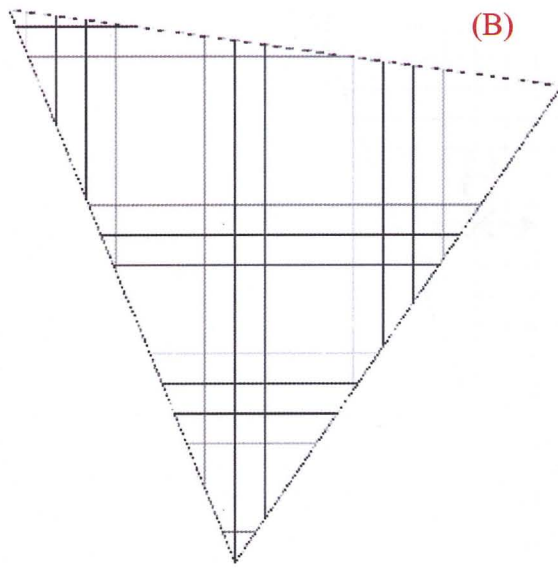
図 4.7 樹の部分の拡大図
Fig. 4.7 The zoom in image of the tree

実験 2 は人工画像を使った提案手法と従来法の比較である。これらの処理は図 4.8 (A)に示される人工画像を 30%(296×296)までにサイズを縮小するものである。説明の都合上、元画像を白(255)と黒(0)の二値だけで説明するが、多値の場合でも処理は一緒である。画像の中に描かれている線の太さは全部 1 画素であるように設定している。この画像を従来法と提案手法で処理した結果を図 4.8(B)から図 4.8(F)に示される。図 4.8(B)は最近傍法の処理結果であり、図 4.8(C)は双線型法の処理結果である。図 4.8(D)は双三次法(B-Spline)の処理結果であり、図 4.8(E)は面積平均法の結果であり、図 4.8(F)(5×5 構造)は顕著度による面積平均法の処理結果である。処理結果を見て分かるように静的なサンプリング範囲を使っている最近傍法、双線型法と双三次法は画像の中のグリッドの形すら保つことが出来なかった。それに対して、動的なサンプリング範囲を利用する面積平均法と提案手法は画像 4.8(E)(F)に示されているように全体的な形を正しく保持できている。

図 4.8(E)と図 4.8(F)の×のところを注目して、矢印の方向で画素の処理結果を比べて見る。点 A(190, 116)を起点に水平方向で左から右への画素値を調べると、元画像の太さが 1 画素であった(255) → (255)→(0) →(255) →(255)の線は面積平均法で処理したら、結果は (255) → (255)→(189) →(237) →(255)となって、縮小処理の過程で太さは 2 画素となっている。提案手法で同じ所をスキャンすると、その結果は(255) → (255)→(162) →(255) →(255)であり、線の太さに変化がない。また、同様に点 B(86, 113)を調べると、面積平均法の結果は(255) → (255)→(209) →(220) →(255)となり、提案手法の結果は(255) → (255)→(165) →(255) →(255)となっている。この結果から分かるように、提案手法は線の太さを保てるほか、線の色も従来の面積平均法より元画像の色(0)に近い。

(A)





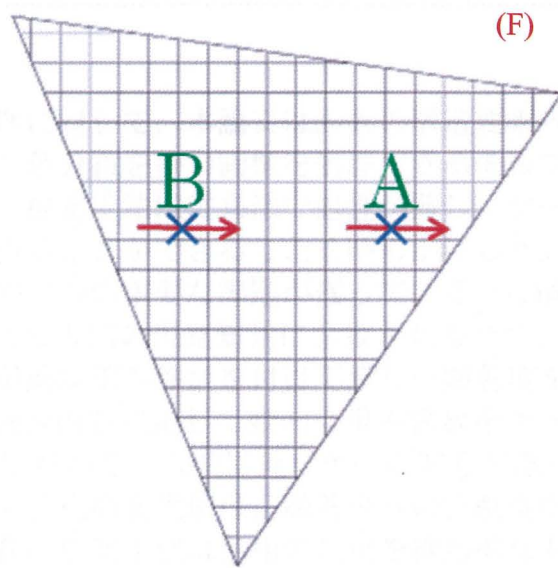


図 4.8 人工画像による縮小処理

Fig. 4.8 The reduction process on artificial image

5. まとめ

モルフォロジー処理に対して、本論文は画素の構造要素内における顕著度をベースにして、複雑な色分布を持つ画像を処理できるモルフォロジー処理法を提案した。実験では、提案手法は従来法で処理が難しいケースでも安定的かつ効果的に処理結果を得ることができることを明らかにした。提案手法をさらに改善するために、将来二つの重要な課題が残っている。(1)画像における画素の「顕著」の度合いをより人間の視覚知覚に合致させるため、修正された(H, S, V)のそれぞれの成分に最適な重み係数を付けて新しい顕著度を重みつけノルムで計算すること。また画像内容に応じて動的に重み係数をチューニングすること。(2)可視光範囲内の色に対して、人間は違う色への関心が違うことが知られている。どの領域の色相の変化値は、顕著度への貢献度が高いかを確認していく必要がある。現在の提案手法は色相の変化を線形的なものと仮設したが、実際、そのような色の変化はその数値と微妙にずれることがある。例えば、赤色のゾーンは0.2ぐらいの広い領域を持つが、その領域内の色はあまり変わらないにたいして、水色などの色では、領域角度範囲は狭いため、色の変化は比較的激しい。顕著度をもっと適切に定義するために、視覚心理学の知見を取り入れて更なる研究が必要であると思われる。

離散的なデジタル画像のスケーリング変換において、従来の最近傍法や線形補間、スプライン補間法では、画素を割り増ししたり、間引いたりする問題が発生する。面積平均法で限なく全画像スペースで問題を克服すると、オーバー平均による画像ぼやけが発生する。また、面積平均法では、本論文は全画像の情報をサンプリングし、あらゆる画素の情報を新画像の計算に入れる顕著度による面積平均法を提出した。高い倍率の縮小過程における部分画素の情報のロスにより発生する違和感を避けた。更に、画像の特徴である高周波数の部分のデータを従来法より多く保存することを成功し、画像の鮮明さも従来法より上がった。提案手法は全画素をサンプリングする高い倍率の縮小アルゴリズムにおいて、結果画像の滑らかさと鮮明さの間でバランスを取る結果となっている。そのバランスをコントロールするパラメータ n (式(5.8))と画像内容の顕著度分布及びスケーリング倍率の関係性はまだ判明していない。現在の n は直接スケーリング倍率と同じ値にさせているが、極端な画像内容やスケーリング倍率でそのアルゴリズムの効果を低レベルに抑えることもある。例えば大きな白い色の写真の真ん中に、一つ画素の大きさの黒い点があるとする。その画像を10倍以上縮小すると、黒い画素の情報は完全に周りの画素に被られることになってしまう。スケーリング倍率に応じて顕著度から画素の重みをコントロールする n の関数をより良いものにすることが将来の課題である。

参考文献

- [1] X.Z.Sun, A.N.Venetsanopoulos:“Nonlinear point operators for edge detection”, Electrical Engineering Journal, Canadian, Vol.11, Issue 3, pp.90-103(1986)
- [2] Vincent Rorre, Tomaso A.Poggio: “On Edge Detection”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.PAMI-8, Issue 2, pp.147-163(1986)
- [3] G. Matheron:“Random Sets and Integral Geometry”, New York: Wiley (1975)
- [4] J. Serra: “Image Analysis and Mathematical Morphology”, New York: Academic (1982)
- [5] N.Bouayanaya, M.Charif-Chefchaoui, D.Schonfeld: “Theoretical Foundations of Spatially-Variant Mathematical Morphology Part I: Binary Images”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.30, Issue 5, pp. 823-836 (2008)
- [6] N.Bouayanaya, D.Schonfeld: “Theoretical Foundations of Spatially-Variant Mathematical Morphology Part II: Gray-Level Images”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.30, Issue 5, pp. 837-850 (2008).
- [7] Suresh Gautam, SukumarM.Brahama:“Overview of Mathmatical Morphology in Power Systems—A Tutorial Approach”, Power &Energy Society General Meeting, 2009. PES '09. IEEE, pp.1-7, (2009)
- [8] 浅野晃, 延原肇: “マセマテイカルモルフォロジーの基礎と新展開”, 電子情報通信学会誌 Vol 92, No.10, pp.876-880 (2009)
- [9] O.Lezirat, A.Elmoataz, C.Meurie:“Mathematical Morphology in Any Color Space”, Image Analysis and Processing Workshops, ICIAPW, pp.183-187 (2007)
- [10]M.C.d'Ornellas, J.A.T.Borges da Costa:“Color Mathematical Morphology Based on Partial Ordering of Spectra”, Computer Graphics and Image Processing, SIBGRAPI,pp.37-44 (2007)
- [11]XuanGuo, BaopingGuo:“Color image morphology based on distances in the HSI color space”, Computing, Communication, Control, and Management, CCCM , pp.264-267 (2009)
- [12]C.M.Cook, G. and E.Delp: “An Investigation of the Use of High Performance Computing for Multiscale Color Image Smoothing Using Mathematical Morphology”, Proc. of the SPIE Conference On Image Modeling, Vol.1904, pp.104-114 (1993)
- [13]A.Hanbury, J.Serra:“Mathematical Morphology in the HLS Colour Space”, Proc. of the British Machine Vision Conference 2001, BMVA, pp.451-460 (2001)
- [14]Thierry Blu, Philippe Thevenaz, Michael Unser: “Linear Interpolation Revitalized”,

IEEE Trans. on Image Processing, Vol.13, Issue 5, pp.710-719(2004)

[15] Thomas M. Lehmann, Claudia Gonner, Klaus Spitzer: "Addendum: B-spline Interpolation in Medical Image Processing", IEEE Trans. on Medical Imaging, Vol.20, Issue 7, pp.660-665(2001)

[16] Hsieh S. Hou, Harry C. Andrews: "Cubic Splines for Image Interpolation and Digital Filtering", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol.26, Issue 6, pp.508-517(1978)

[17] Robert G. Keys: "Cubic Convolution Interpolation for Digital Image Processing", IEEE Trans. on Acoustics, Speech and Signal Processing, Vol.29, Issue 6, pp.1153-1160(1981)

[18] Zaifeng Shi, Suying Yao, Yingchun Zhao: "A Novel Video Image Scaling Algorithm based on Morphological Edge Interpolation", 2008 International Conference on Neural Networks and Signal Processing, pp.388-391(2008)

発表リスト

口頭発表：

[1] Lei Sheng, Shanjun Zhang, Hongbing ZHU, "The Design of an Optical Lens Simulation by OpenGL", The 2nd International Conference on Intelligent Networks and Intelligent Systems(ICINIS 09), Nov.1-3, 2009, Tianjin, China.

[2] 大橋智樹, 盛 磊, 張 善俊, 「オートステレオグラムにおける変則的融合の検証」, FIT2011 第10回情報科学技術フォーラム, 2011年9月7日~9日, 函館大学.

[3] 大橋智樹, 盛 磊, 張 善俊, 「オートステレオグラムにおける融像とずれ幅の関係性」, 2011年ソサイエティ大会, 2011年9月13日~16日, 北海道大学.

[4] Lei Sheng, Kazuyoshi YOSHINO, Shanjun ZHANG, "Morphology Algorithm using Conspicuousness Map for Color Image Processing", The 5th International Conference on Intelligent Networks and Intelligent Systems(ICINS 12), Nov.1-3, 2012, Tianjin, China.

[5] 盛磊, 張善俊, 和芳吉野, 「カラー画像の顕著度マップによるモルフォロジー処理法」, 平成25年度電気・情報関係学会北海道支部連合大会, 2013年10月19日~20日, 室蘭工業大学.

[6] Yi Hu, Lei Sheng, Shanjun Zhang "A Study of Calligraphy Generation Method

Based on Handwriting", 平成25年度電気・情報関係学会北海道支部連合大会, 2013年10月19日～20日, 室蘭工業大学.

査読付き論文：

[1] Lei Sheng, Shanjun Zhang, Hongbing Zhu, "The Design of an Optical Lens Simulation by OpenGL", International Journal of Intelligent Engineering & Systems, Vol.2, No.4, pp.27-34, Dec.2009.

[2] 張善俊, 盛磊, 「画像をコードブックに利用する投票の暗号化方法」, 画像電子学会誌, Vol.40, No.1, pp.208 -216, Jan.2011.

[3] 盛磊, 和芳吉野, 張善俊, 「カラー画像の顕著度マップによるモルフォロジー処理法」, 画像電子学会誌(2014年3月掲載予定).

[4] Yi Hu, Lei Sheng, Shanjun Zhang, "Design of Continuous Indoor Navigation System Based On INS and Wifi", Applied Mechanics and Materials, Vol.303-306, pp 2046-2049, Feb.2013.