

## 携帯機器による高解像度画像観賞のための画像処理技術の開発

代表研究者	井 上 光 平	九州大学大学院芸術工学研究院助教
共同研究者	浦 浜 喜 一	九州大学大学院芸術工学研究院教授
〃	原 健 二	九州大学大学院芸術工学研究院准教授

### 1 はじめに

近年、携帯電話をはじめとする各種携帯機器の小型化・軽量化が進み、どこでも簡単に必要な情報を見ることができるようになり、携帯機器の画面で高解像度の画像や映像を十分な画質で視聴したいという要求も高まってきている。そこで本研究では、携帯機器などの小さな画面で画質を損なわずに高解像度の静止画や動画を観賞するための画像処理技術を開発することを主な目的として、いくつかの画像処理アルゴリズムを提案し、その有効性を実験により検証した。以下に、本研究の概要を報告する。

### 2 パノラマ画像の効率的表示法

横長の長方形のパノラマ画像が与えられるとき、局所的カラーヒストグラムのエントロピーを画像の横方向に均等化することによって、相対的にエントロピーの高い領域を広め、低い領域を狭めながらパノラマ画像の横幅を縮小する方法を提案する。Avidanらの seam carving 法[1]との比較実験を行い、提案手法によって高速にパノラマ画像の横幅を縮小することができることを示す。また、エントロピーを強調することによって非線形性を強めたパノラマ画像の横幅縮小法も示す[2, 3]。

#### 2-1 エントロピー均等化によるパノラマ画像の横幅縮小

パノラマ画像が1枚与えられるとし、それを  $f = [f_{ij}], i = 1, \dots, m; j = 1, \dots, n$  とする。  $f_{ij} = [r_{ij}, g_{ij}, b_{ij}]$  は画素  $(i, j)$  のRGB値を要素とするカラーベクトルである。  $m, n$  はそれぞれ縦、横の画素数であり、  $m < n$  とする。本稿では、  $\tilde{n} < n$  が与えられたときに、  $f$  の横幅を縮小した画像  $\tilde{f} = [\tilde{f}_{ik}], i = 1, \dots, m; k = 1, \dots, \tilde{n}$  を生成する方法を提案する。

まず、  $f$  を  $C$  色に減色し、画素  $(i, j)$  の色番号を  $c_{ij} \in \{1, \dots, C\}$  とする。画像の各列  $j$  で局所的なカラーヒストグラム

$$h(c, j) = \sum_{i=1}^m \sum_{l=-L}^L \delta(c, c_{i, j+l}) e^{-\alpha l^2} \cdots (1)$$

を計算する。ここで  $\delta(c, c_{i, j+l})$  は、  $c = c_{i, j+l}$  のとき1であり、その他は0である。  $e^{-\alpha l^2}$  の値は  $l$  が0に近いほど1に近付き、遠いほど0に近づく。これによって  $j$  の近傍の色を反映した局所的カラーヒストグラムが求められる。  $\alpha$  は局所性を調節するための正定数である。計算量を削減するために、  $l$  の変域を  $e^{-\alpha l^2} \geq \varepsilon$  が成り立つ範囲に制限する。この範囲を  $[-L, L]$  とすると、  $L$  は  $e^{-\alpha L^2} = \varepsilon$  より、  $L = \text{round}(\sqrt{-(\ln \varepsilon)/\alpha})$  となる。ここで  $\text{round}$  は最も近い整数への丸めを表し、  $\varepsilon$  は微小な正定数である。次に  $p(c, j) = h(c, j) / \sum_{c=1}^C h(c, j)$  とし  $\sum_{c=1}^C p(c, j) = 1$  と規格化する。各  $j$  についてエントロピー

$$e(j) = -\sum_{c=1}^C p(c, j) \ln p(c, j) \cdots (2)$$

を計算し、  $\tilde{e}(j) = (e(j) - e_{\min}) / (e_{\max} - e_{\min})$  とする。ここで  $e_{\min} = \min\{e(1), \dots, e(n)\}$ ,  $e_{\max} = \max\{e(1), \dots, e(n)\}$  である。  $\tilde{e}(j)$  を累積したものを  $\tilde{E}(j) = \sum_{j'=1}^j \tilde{e}(j')$  とする。すなわち、各  $j$  について、  $\tilde{e}(1), \dots, \tilde{e}(j)$  を足し合わせた値を  $\tilde{E}(j)$  とする。  $\tilde{E}(j)$  は単調増加関数であり、最小値は  $\tilde{E}(1)$ 、最大値は  $\tilde{E}(n)$  である。この  $\tilde{E}(1), \tilde{E}(n)$  をそれぞれ横幅縮小画像の左端の横座標1、右端の横座標  $\tilde{n}$  に対応付ける。すなわち

$$k(j) = (\tilde{n} - 1) \frac{\tilde{E}(j) - \tilde{E}(1)}{\tilde{E}(n) - \tilde{E}(1)} + 1 \cdots (3)$$

によって  $f$  の第  $j$  列を  $\tilde{f}$  の第  $k(j)$  列に対応付ける．実際には， $j$  から  $k(j)$  への対応付けは多対 1 となる場合があるので，同じ列  $k$  に対応付けられる列  $j$  の平均

$$\tilde{f}_{ik} = \frac{\sum_{j,k(j)=k} f_{ij}}{\sum_{j,k(j)=k} 1}, \quad i=1, \dots, m \quad \dots (4)$$

を横幅縮小画像の第  $k$  列とする．

パノラマ画像の例を図 1 に示す．画像サイズは，図 1(a) が  $m = 270, n = 1000$  であり，図 1(b) が  $m = 270, n = 2200$  である．アスペクト比はそれぞれ 3.7 と 8.1 である．

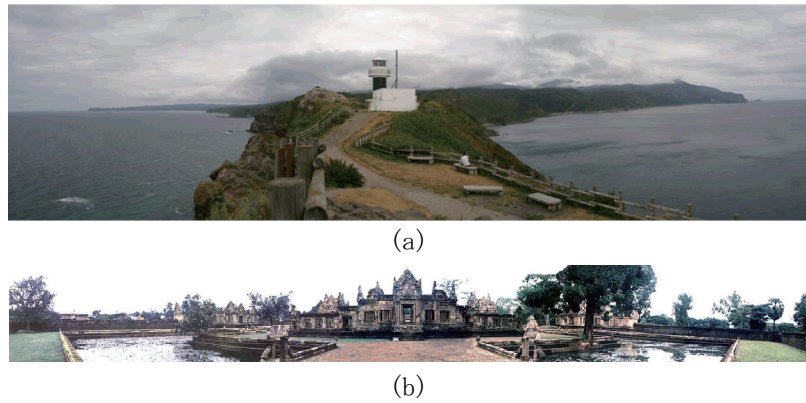


図 1: パノラマ画像の例

$\tilde{n} = 4m/3 = 360$  として図 1(a), (b) から求めた  $k(j)$  をそれぞれ図 2(a), (b) に示す．

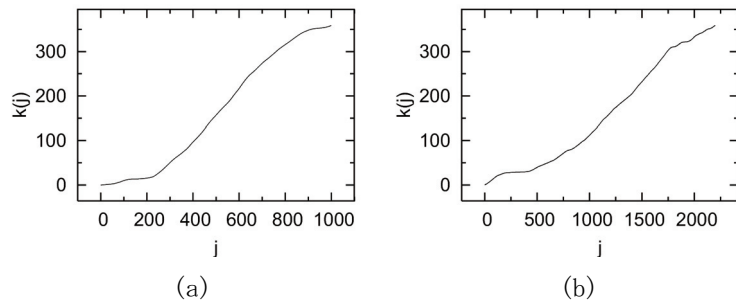


図 2: 図 1(a), (b) の画像の  $k(j)$

横軸は  $j$  であり，縦軸は  $k(j)$  である．グラフの傾きが相対的に小さい領域は横幅が縮小され，大きい領域は拡大される．なお， $k(j)$  は，(1) 式で  $i$  についての総和をとっていることからわかるように，すべての  $i$  (縦方向の座標) で共通である．横幅の縮小結果を図 3 に示す．



図 3: 図 1(a), (b) の横幅縮小画像

図 3(a) では，半島の両側の海と空の領域が縮小され，半島の領域に広い領域が割当てられている．図 3(b)

では、建物や樹木のない領域が縮小され、中心部の建物や樹木の領域が拡大されている。この実験では  $C=256$  とした。また  $\alpha=\varepsilon=0.01$  とし、このとき  $L=21$  であった。比較のために、Avidan らの seam carving 法[1]による結果を図4に示す。



図4: seam carving 法[1]による図1(a), (b)の横幅縮小画像

図4(a)では、図3(a)よりも半島の領域が広く残っており、効率的な横幅縮小が行われているといえる。一方、図4(b)では、樹木や建物間の空の領域が削除されて、隣り合う樹木や建物がくっついてしまい、画像内容がわかりにくくなっている。このように、seam carving 法[1]では、比較的アスペクト比の小さい図1(a)のようなパノラマ画像では良好な結果が得られるが、アスペクト比の大きい図1(b)のようなパノラマ画像では良好な結果が得られない場合もある。seam carving 法では、画像の上辺から下辺に至る8連結のパスの中からエネルギー関数を最小化するものを選び、選ばれたパスを削除するという操作を繰り返しながら画像の横幅を縮小していく。したがって、削除されない重要な領域は、元の大きさのまま縮小画像内に残ることになる。重要な領域が縮小画像の横幅よりも大きいときには、重要な領域も徐々に削除されていき、元のパノラマ画像の内容が失われていく。それに対して、提案手法では、画像のある部分を削除するのではなく、画像の各列でエントロピーによって縮小率を調節して横幅を縮小するので、重要な領域は元の大きさよりは縮小されるが、縮小画像内では相対的に広い領域が割当てられ、元のパノラマ画像の内容が効率的に保存される。

計算時間を図5に示す。横軸は横幅縮小画像の横幅の画素数であり、縦軸はCPU時間である。実線は seam carving 法[1]であり、破線は提案手法である。seam carving 法では、横幅の画素数が少ないほど計算時間が長くなるが、提案手法では計算時間はほぼ一定であり、seam carving 法よりも高速である。計算環境はPentium 4 CPU 3.40GHz, 2.00GB RAM であり、プログラミング言語はMicrosoft Visual C++である。

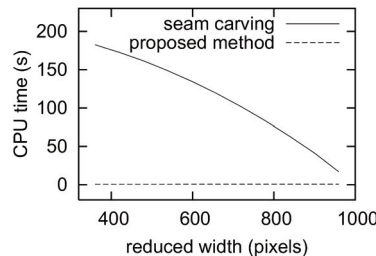


図5: 計算時間

## 2-2 エントロピーの強調

前節では、(2)式のエントロピーに基づいて、パノラマ画像の横幅を非線形に縮小したが、ここでは、エントロピーを強調することによって、更に非線形性を強めた横幅縮小法を示す。

エントロピーの違いを強調するために、(2)式の代わりに

$$e(j) = \left( - \sum_{c=1}^C p(c, j) \ln p(c, j) \right)^r \cdots (5)$$

を用いる。この式は、 $r$ をパラメータとして(2)式を拡張したものになっており、 $r=1$ のとき(2)式に一致し、 $r>1$ のときエントロピーの違いが強調される。

この方法で、図1(a), (b)のパノラマ画像から  $k(j)$  を求めた結果をそれぞれ図6(a), (b)に実線で示す。 $r=10$  とした。また、縮小後の横幅の画素数  $\tilde{n}$  は図3と同じく  $\tilde{n}=4m/3=360$  とした。図6(a), (b)の破線は、 $r=1$ で

$k(j)$  であり、それぞれ図 3(a), (b) の実線と同じである。両者(実線と破線)を比較すると、エントロピーを強調することによって、曲線の非線形性が強くなっていることがわかる。エントロピーの強調によって、エントロピーの小さい領域では、 $k(j)$  の傾きがより小さくなり、エントロピーの大きい領域では、 $k(j)$  の傾きがより大きくなっている。その結果、横幅縮小画像では、エントロピーの小さい領域は相対的により縮小され、エントロピーの大きい領域はより拡大される。

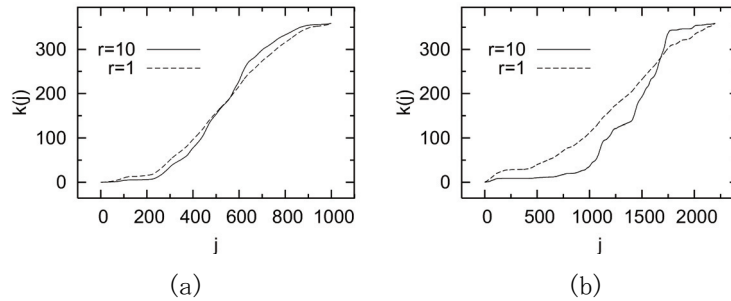


図 6: エントロピー強調法による図 1(a), (b) の画像の  $k(j)$

この方法による横幅縮小画像を図 7 に示す。図 7(a) では、図 3(a) と比較すると、中央付近の灯台の領域が広がっており、図 4(a) の seam carving 法[1]による結果に近い画像が得られた。図 7(b) では、図 3(b) と比較すると、中心部の建物とその右横の樹木の領域が更に広くなり、その他の領域は更に狭くなっているが、図 4(b) の seam carving 法[1]による結果のように、建物や樹木が入り乱れて画像内容がわかりにくくなるということではなく、建物や樹木の大きな形は保たれている。

エントロピーの強調を行う方法と行わない方法の違いは、エントロピーの  $\gamma$  乗の有無だけであるので、計算時間はどちらもほぼ同じであり、seam carving 法[1]よりも高速である。



図 7: エントロピー強調法による図 1(a), (b) の横幅縮小画像

他の画像例を図 8 に示す。図 8(a) のパノラマ画像の横幅を  $r=1, 10$  で縮小した結果をそれぞれ同図(b), (c) に示す。図 8(a) の画像サイズは  $m=720, n=2873$  であり、同図(b), (c) では  $\tilde{n}=4m/3=960$  とした。右側の高いビルやタワーが立ち並んでいる領域が相対的に広げられていることがわかる。また、図 8(c) では、ビルやタワーの横幅が、同図(b)よりも広がっている。



図 8: 他の画像例: (a) パノラマ画像, (b) 横幅縮小画像 ( $\gamma=1$ ), (c) 横幅縮小画像 ( $\gamma=10$ )

### 3 ウォークスルーによる写真の小画面映像表示

大サイズの画像を携帯電話などの小画面に表示するとき、単純に縮小すると細部が視認できなくなる。細

部を縮小せずに表示する手法として、大画像のなかの主要領域を次々に巡視するような小フレームサイズの動画に変換する方法がある。ここでは、画像から注目点を抽出し、それらの点を通る最短ハミルトン経路に沿ってパンしながら各注目点をズームしつつ表示する方法を提案する。注目度画像から注目点を抽出し、それらを経由する最短ハミルトン経路を巡回セールスマン問題の近似解法を利用して求める。各注目点での拡大縮小率は注目度に基づいて決める。この画像表示法を多重画像にも応用し、各注目点において最も注目度が高い成分画像を順次表示する手法を示す[4]。

### 3-1 注視点の抽出と巡視経路の生成

まず、Achanta らの注目領域検出法[5]を用いて注目度画像を生成し、以下の手順で注目点を抽出する。

ステップ 1: 注目度画像のなかで値が最も大きい画素を探して注目点のリストに入れる。

ステップ 2: リストに入れた注目点を中心とするウィンドウ内の画素値を 0 にした画像を新たに注目度画像とする。

ステップ 3: 注目度画像のなかで値が最も大きい画素を探し、その点の注目度がしきい値以上なら、その点を注目点のリストに加え、ステップ 2 へ行く。注目度がしきい値未満なら終了。

次に、抽出した全ての注視点を巡視する経路を求める。人の視点追跡のシミュレーション[6]では注目度の順番に注視点を巡ることが多いが、ここでの目的は抽出された全ての注視点を効率的に巡視することであるから、全注視点を通る最短ハミルトン経路に沿って表示する。その際、Kim らの超解像法[7]を用いて注目領域を拡大縮小したり、移動速度を変えたり、経路をスプライン補間によって滑らかにするなどの改良を施した。図 9 に抽出した注視点と巡視経路の例を示す。

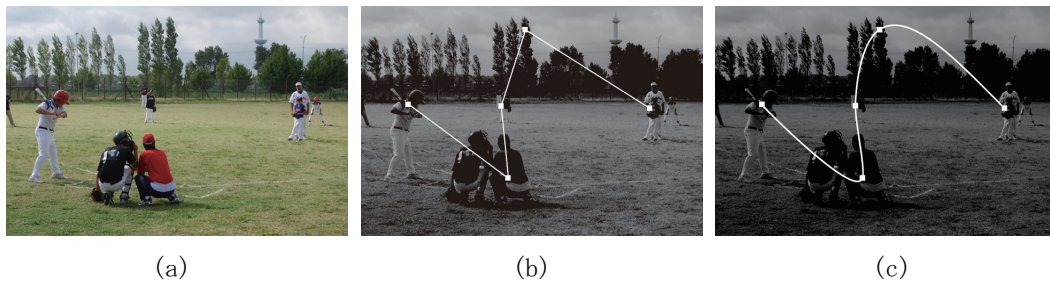


図 9: 注視点と巡視経路。(a) 入力画像, (b) 最短ハミルトン経路, (c) スプライン補間による経路

### 3-2 顔検出の併用

以上では、注目点は Achanta[5]の注目度画像だけから検出した。しかし、Achanta の方法[5]では人物の顔は必ずしも注目度が高くなるとは限らない。人が写真を見るときには、人物の顔には注目するので、従来の画像表示法[8]–[11]と同様に、注目度計算とは別に顔検出も行い、両者を組み合わせて注目点を抽出するのも試みた。図 10 に顔検出を併用した例を示す。図 10(a)の顔検出結果が同図(b)であり、全員の顔が検出されている。(c)が最短ハミルトン経路、(d)がスプライン補間である。

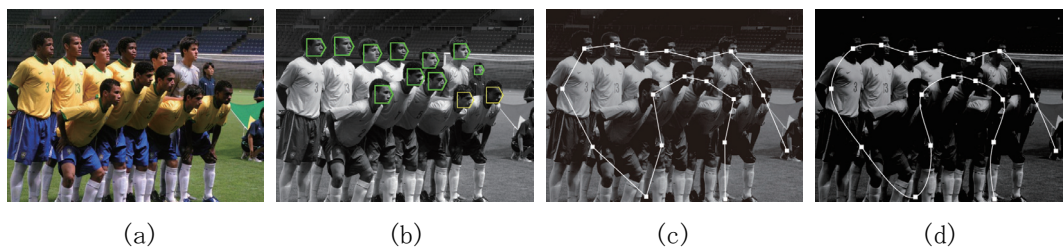


図 10: 顔検出併用の例

### 3-3 多重画像への拡張

以上では入力画像は 1 枚だけであるが、複数枚ある場合にも拡張した。ここでは、高ダイナミックレンジ(HDR)画像の作成に利用されるような多重露光画像を取り上げ、そのような複数枚の画像の情報が把握できるような動画を生成した。生成の手順は以下の通りである。

まず最初に、各成分画像ごとに注目度画像を生成する。次に、その 2 枚の注目度画像の各画素での画素値

の大きい(すなわち注目度が高い)ほうを画素値とする画像を作り，その画像に対して3-1節の手順で注目点を抽出して，描く注目点での拡大縮小率を求める．次に，この注目点全てを巡る最短ハミルトン経路を求める．最後に，この経路に沿ったウォークスルー動画を生成する．各注目点では，注目度が高いほうの成分画像を表示する．注目点から注目点への移動において，両注目点の選択画像が異なるときには，中間点から両注目点への距離の内分率で両成分画像をブレンドする．このようにすると，いつでも注目度が高い成分画像が表示され，各領域が最も見やすい露光の画像が自動的に選ばれて表示されると思われる．この画像表示法も1種の画像融合法になっている．実際の多重露光画像を用いた実験では，各注目点において露光が適度な画像が選ばれており，見やすい動画が得られた．

#### 4 連結リストを用いた Dynamic Mean Shift による画像のセグメンテーション

局所的モード探索の代表的手法の1つである mean shift (MS) [12, 13]の高速化法として, dynamic MS (DMS) [14, 15]が提案されている．DMS では，データを動的に更新しながら重なったデータを削除することによって計算量を削減するが，更新の初期段階では，データはほとんど削減されておらず，従って MS と同程度の計算量を要する．実際，DMS の計算量の大半はこの初期段階に集中しており，画像やビデオなどの大規模データを処理するのは MS と同様に困難な場合が多い．ここでは，DMS による画像のセグメンテーションを高速化する方法として，連結リストを用いた DMS を提案する．各画素に連結リストを用意し，更新されたデータをその最近傍画素の連結リストに格納していき，画像上でのデータの近傍関係を用いて DMS の計算量を削減する．本手法によって，DMS とほぼ同様のセグメンテーション結果が DMS よりも短時間で得られることを実験で確認する [16]．

図 11(a)に示すモノクロ画像を用いて実験を行った．



図 11: セグメンテーションの例: (a) 入力画像, (b) DMS, (c) DMS-L

画素数は  $256 \times 256$  である．従来の DMS と提案手法(DMS-L と略記する)によるセグメンテーションの結果をそれぞれ図 11(b), (c)に示す．これら 2 枚の画像のピーク信号対雑音比 (PSNR) は 43.94 (dB) であり，ほぼ同様の結果が得られているといえる．計算時間は，DMS が 3048.61 秒，DMS-L が 1079.69 秒である．計算環境は，Pentium(R) 4 CPU 3.40GHz, 2.00GB RAM であり，プログラミング言語は Microsoft Visual C++ である．

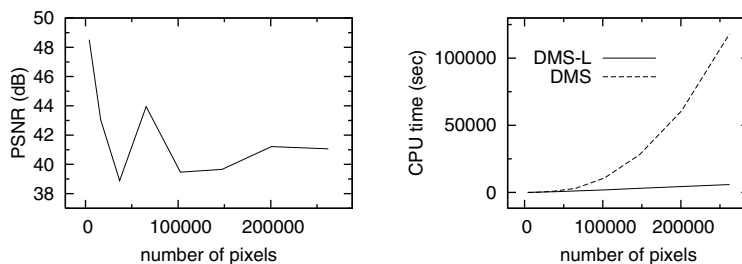


図 12: 異なる画素数での結果: (a) PSNR, (b) 計算時間

図 11(a)の画像の画素数を  $64 \times 64$  から  $512 \times 512$  まで変えたときの PSNR を図 12(a)に示す．横軸は画素数で

あり、縦軸は PSNR である。どの画素数でも PSNR 値が 38 (dB) 以上であり、DMS-L によって、画素数によらずに DMS とほぼ同様のセグメンテーションが得られることがわかる。計算時間を図 12 (b) に示す。横軸は図 12 (a) と同じく画素数であり、縦軸は計算時間である。破線は DMS であり、実線は DMS-L である。画素数が大きいほど DMS-L の高速性が顕著になることがわかる。

## 5 重み付きヒストグラムインタセクション最小化による画像のクリッピング

画像から矩形領域を切り抜く方法の 1 つとして、矩形の内側と外側のカラーヒストグラムのヒストグラムインタセクション (HI) を最小化するような矩形の位置と大きさを求める方法を提案する。従来の HI を重み付きに拡張することによって、矩形内部に主に対象物が含まれ、背景があまり含まれないような切抜きができることを実験で示す [17, 18]。

### 5-1 重み付きヒストグラムインタセクション

限定色画像のカラーヒストグラム (color histogram: CH) を考える。色数を  $C$  とし、 $c$  番目の色の出現頻度 (画素数) を  $\tilde{h}_c$  とし、CH を  $\tilde{h} = [\tilde{h}_1, \dots, \tilde{h}_C]$  とする。 $\tilde{h}$  の要素の総和を 1 に規格化したものを  $h = [h_1, \dots, h_C]$ ,  $h_c = \tilde{h}_c / \sum_{c=1}^C \tilde{h}_c$  とする。画像  $A, B$  の規格化 CH (normalized CH: NCH) をそれぞれ  $h^A = [h_1^A, \dots, h_C^A]$ ,  $h^B = [h_1^B, \dots, h_C^B]$  とすると、これらのヒストグラムインタセクション (histogram intersection: HI) は

$$HI(h^A, h^B) = \sum_{c=1}^C \min\{h_c^A, h_c^B\} \cdot \dots \quad (6)$$

で定義される [19]。ここでは、 $A, B$  のそれぞれについて重み  $w^A = [w_1^A, \dots, w_C^A]$ ,  $w^B = [w_1^B, \dots, w_C^B]$  が与えられるとし、2 つの NCH  $h^A, h^B$  の重み付きの HI (weighted HI: WHI)

$$WHI(h^A, h^B; w^A, w^B) = \sum_{c=1}^C w_c^A w_c^B \min\{h_c^A, h_c^B\} \cdot \dots \quad (7)$$

を考える。筆者らは以前、 $\sum_{c=1}^C w_c \min\{h_c^A, h_c^B\}$  という形の WHI を提案したが [20]、その重み  $w_c$  は  $A, B$  に依存しないのに対して、式 (7) の重みは  $A, B$  に依存するという点で両者は異なっている。

### 5-2 WHI 最小化による画像のクリッピング

ここでは、与えられた画像の部分矩形を切り抜くことを考え、それを画像のクリッピングと呼ぶことにする。縦  $m$  画素、横  $n$  画素の画像を縦  $\tilde{m}$  画素、横  $\tilde{n}$  画素の矩形でクリッピングするとし、矩形の左上の画素の座標を  $(\tilde{i}, \tilde{j})$  とする。画像を矩形の内側と外側の 2 つの領域に分け、各領域の NCH をそれぞれ  $h^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})$ ,  $h^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})$  とする。 $h$  の右肩の  $I$  と  $O$  はそれぞれ矩形の内側 (inside) と外側 (outside) を意味する。ここでは画像のクリッピングを、これら 2 つの NCH の WHI を最小化するような矩形の位置と大きさを求める問題として定式化する：

$$\min_{\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}} WHI(h^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}), h^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}); w^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}), w^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})) \cdot \dots \quad (8)$$

重み  $w^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})$ ,  $w^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})$  の各要素は

$$w_c^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}) = h_c^I(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}) \cdot \dots \quad (9)$$

$$w_c^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n}) = h_c^O(\tilde{i}, \tilde{j}, \tilde{m}, \tilde{n})^{-1} \cdot \dots \quad (10)$$

とする。すなわち、矩形内部の主要な色の重みを大きくし、矩形外部の主要な色の重みを小さくする。

式 (8) の解を求める方法として、全探索と局所探索を行った。その際、インテグラルヒストグラム [21] を利用することによって計算量を削減した。

### 5-3 実験例

実際の画像を用いた実験例を示す。全探索の結果を図 13 に示す。探索によって得られた矩形を白枠で示している。図 13 (a), (b) は従来の重みなし HI による結果であり、非常に小さい矩形が選択されており、昆虫や

自動車といった対象物をクリッピングするのに失敗している．それに対して，図 13(c), (d)は重み付き HI (WHI)による結果であり，対象物を適切に取り囲む矩形が選択されている．

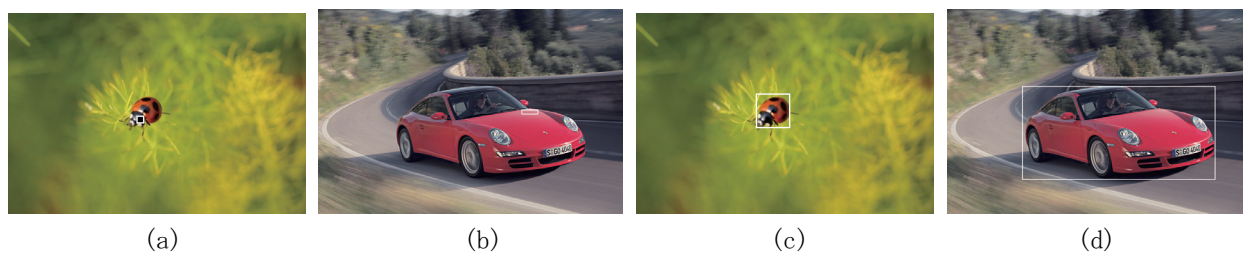


図 13: 全探索の例: (a), (b) 重みなし HI, (c), (d) 重み付き HI

次に，局所探索の結果を図 14 に示す．図 14(a), (b)は探索途中の矩形の更新の様子を表しており，図 14(c), (d)は最終的に得られた矩形領域を拡大表示したものである．対象物の領域が適切にクリッピングされている．また，局所探索では全探索よりも短い計算時間で全探索と同様のクリッピング結果が得られた．

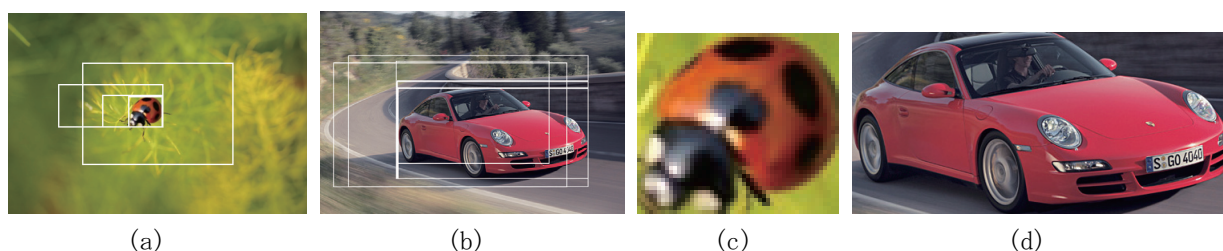


図 14: 局所探索の例: (a), (b) 矩形の更新の様子, (c), (d) クリッピングの結果

次に，複数の矩形領域をクリッピングした結果を図 15 示す．図 15(a)は入力画像である．矩形の数は 2 とし，図 15(b)のように初期配置し，互いの領域が重ならないようにして局所探索を行った．矩形の移動の様子を図 15(c)に示す．クリッピングの結果を図 15(d)に示す．

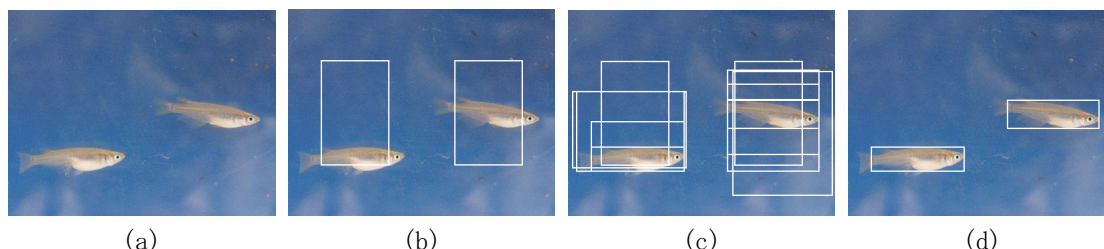


図 15: 複数の矩形領域探索の例: (a) 入力画像, (b) 初期配置, (c) 局所探索の様子, (d) 探索結果

## 6 反復クロスバイラテラルフィルタによる画像の 2 値化

画像の 2 値化は文字や図形の認識の前処理の一つとして古くから研究されているが，最近でもノンフォトリアリスティックレンダリング[22]などに利用されている．ここでは，画像の局所的な明暗の関係を保った 2 値化の方法として，アンシャープマスキングで強調されるエッジの明暗の関係を反復クロスバイラテラルフィルタ (ICBF) で画像全体に伝搬させてから 2 値化する方法を提案する．

図 16 に簡単な画像による例を示す．図 16(a)は入力画像であり，この画像に対するアンシャープマスキングの強調成分は図 16(b)のようになる．図 16(b)に対して図 16(a)を補助画像とする ICBF をかけると図 16(c)のような画像が得られる．それを 2 値化すると図 16(d)のような 2 値画像が得られる．中央の灰色の領域が白と黒の領域に分かれることによってすべての主要なエッジが表現されている．

次に，実際の写真での結果を図 17 に示す．図 17(a)は入力画像であり，これを閾値 128 で単純に 2 値化すると図 17(b)のような 2 値画像が得られる．本手法による結果を図 17(c)に示す．図 17(b)では花の裏側の暗



い部分のエッジが消えているが、本手法による結果(図 17(c))では保たれている。また、花の中央の細かな構造の再現性も図 17(c)のほうが優れているといえる。

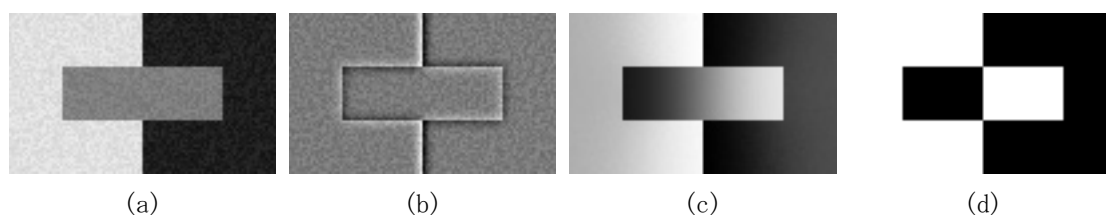


図 16: (a) 入力画像, (b) アンシャープマスキングの強調成分, (c) ICBF の出力画像, (d) 2 値化の結果

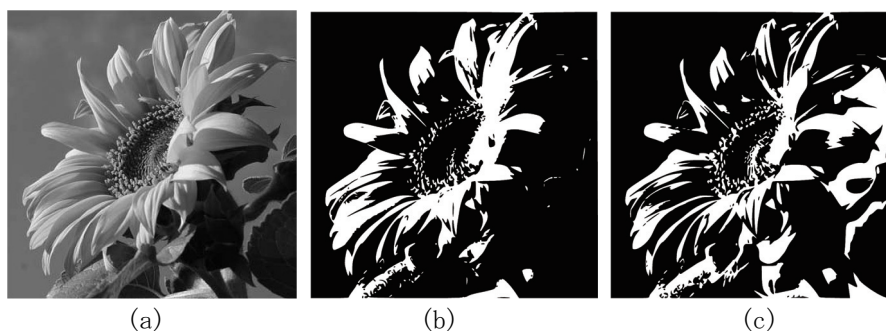


図 17: (a) 入力画像, (b) 閾値 128 による 2 値化, (c) 本手法による 2 値化

## 7 まとめ

本研究では、携帯機器などの小さな画面で画質を損なわずに高解像度の静止画や動画を観賞するための画像処理技術を開発することを目的として、高解像度のパノラマ写真の内容を保存した縮小法、高解像度画像の小画面映像表示法、画像のクリッピング法を提案した。また、高解像度画像処理の基本手法として、連結リストを用いた dynamic mean shift の高速化法と画像の局所的な明暗の関係を保った 2 値化法を提案した。

画像やビデオの内容に応じたリサイジングやリターゲットングの研究は近年、活発化の傾向にあり、今年度も新手法が多数提案された。それらの手法の比較検討と改良が今後の課題である。

## 【参考文献】

- [1] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," ACM Trans. Graphics, vol. 26, no. 3, SIGGRAPH 2007.
- [2] 松島祐輔, 井上光平, 浦浜喜一, "エントロピー均等化によるパノラマ画像の横幅縮小--パノラマ画像の内容を効率的に表示する一手法," 画像ラボ, vol. 20, no. 5, pp. 7-11, May 2009.
- [3] 井上光平, 原健二, 浦浜喜一, "エントロピー強調によるパノラマ画像の効果的横幅縮小," 映像情報メディア学会誌, vol. 63, no. 11, pp. 1622-1624, Nov. 2009.
- [4] 弓場達也, 井上光平, 原健二, 浦浜喜一, "ウォークスルーによる写真の小画面映像表示," 画像の認識・理解シンポジウム (MIRU), IS2-34, pp. 1081-1088, Jul. 2009.
- [5] R. Achanta, F. J. Estrada, P. Wils and S. Susstrunk, "Salient region detection and segmentation," Proc. ICVS, pp. 66-75, 2008.
- [6] L. Itti, C. Koch and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," IEEE Trans. Patt. Anal. Mach. Intell., 20, pp.1254-1259, 1998.
- [7] K. I. Kim and Y. Kwon, "Example-based learning for single-image superresolution," Proc. DAGM, pp.456-465, 2008.
- [8] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H. Zhang, H.-Q. Zhou, "A visual attention model for adapting images on small displays," Multimedia Syst. 9, 4, pp.353-364, 2003.
- [9] X. Xin, H. Liu, W.-Y. Ma and H.-J. Zhang, "Browsing large pictures under limited display sizes," IEEE Trans. Multimedia, 8, 4, pp.707-714, 2006.

- [10] P. Pinho, J. Baltazar and F. Pereira, "Integrating low-level and semantic visual cues for improved image-to-video experiences," Proc. ICIAR, pp.832-843, 2006.
- [11] S. Jiang, H. Liu, Z. Zhao, Q. Huang and W. Gao, "Generating video sequence from photo image for mobile screens by content analysis," Proc. ICME, pp.1475-1478, 2007.
- [12] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," IEEE Trans. PAMI, vol. 24, no. 5, pp. 603-619, 2002.
- [13] 岡田和典, "ミーンシフトの原理と応用 (Tutorial Lecture)," 信学技報, vol. 107, no. 539, PRMU2007-308, pp. 401-414, Mar. 2008.
- [14] K. Zhang et al., "Accelerated Convergence Using Dynamic Mean Shift," Proc. ECCV, LNCS, vol. 3952, pp. 257-268, 2006.
- [15] M. A. Carreira-Perpinan, "Fast nonparametric clustering with Gaussian blurring mean-shift," Proc. ICML, pp. 153-160, 2006.
- [16] 井上光平, 浦浜喜一, "連結リストを用いた Dynamic Mean Shift による画像のセグメンテーション," 映像メディア処理シンポジウム (IMPS), pp. 13-14, Oct. 2009.
- [17] 井上光平, 原健二, 浦浜喜一, "重み付きヒストグラムインタセクション最小化による画像のクリッピング," 信学技報, vol. 109, no. 292, IE2009-113, pp. 85-88, Nov. 2009.
- [18] K. Inoue, K. Hara, and K. Urahama, "Image Clipping by Weighted Histogram Intersection Minimization," 16th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV), pp. 57-61, Feb. 2010.
- [19] M. J. Swain and D. H. Ballard, "Color indexing," International Journal of Computer Vision, vol. 7, pp. 11-32, 1991.
- [20] 井上光平, 浦浜喜一, "ヒストグラムインタセクションの上限と下限に基づくフィルタリング検索," 信学論, vol. J86-D2, no.2, pp. 368-370, Feb. 2003.
- [21] F. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," Proc. CVPR, vol. 1, pp. 829-836, Jun. 2005.
- [22] J. Xu and C.S. Kaplan, "Artistic thresholding," Proc. NPAR, pp. 39-47, 2008.
- [23] 井上光平, 浦浜喜一, "反復クロスバイラテラルフィルタによる画像の2値化," 2009年映像情報メディア学会冬季大会, 9-2, Dec. 2009.

### 〈発表資料〉

題名	掲載誌・学会名等	発表年月
エントロピー均等化によるパノラマ画像の横幅縮小--パノラマ画像の内容を効率的に表示する一手法	画像ラボ	2009年5月
ウォークスルーによる写真の小画面映像表示	画像の認識・理解シンポジウム	2009年7月
連結リストを用いた Dynamic Mean Shift による画像のセグメンテーション	映像メディア処理シンポジウム	2009年10月
エントロピー強調によるパノラマ画像の効果的横幅縮小	映像情報メディア学会誌	2009年11月
重み付きヒストグラムインタセクション最小化による画像のクリッピング	電子情報通信学会画像工学研究会	2009年11月
反復クロスバイラテラルフィルタによる画像の2値化	映像情報メディア学会冬季大会	2009年12月
Image Clipping by Weighted Histogram Intersection Minimization	16th Korea-Japan Joint Workshop on Frontiers of Computer Vision	2010年2月