

シームカービングアルゴリズムの FPGA 実装

木幡 康秀† 風間 萌美‡ 八木澤 良文‡ 出崎 善久‡

†茨城大学大学院理工学研究科メディア通信工学専攻

‡茨城大学工学部メディア通信工学科

1. はじめに

携帯端末等に搭載された多様なディスプレイの普及に伴って、画像のリサイズに対するニーズが高くなっている。画面の端を単純に削除するクロッピング (cropping), リサイズ後のサイズに合わせて画像全体を線形変換するスケーリング (scaling) に代表される従来のリサイズ手法では、人や物等の画像中の重要なオブジェクトに顕著な影響を与えるという問題がある (図 1(b), (c))。このため、オブジェクトの形状を保存したままリサイズを行うアルゴリズム (content aware resizing) の研究が近年盛んになっている。

S. Avidan らによって提案されたシームカービング (seam carving) は、このようなアルゴリズムの一つである [1]。シームカービングでは、背景等の重要度が低い部分のみを取り除き、重要な部分をそのまま残すことが可能である (図 1(d))。

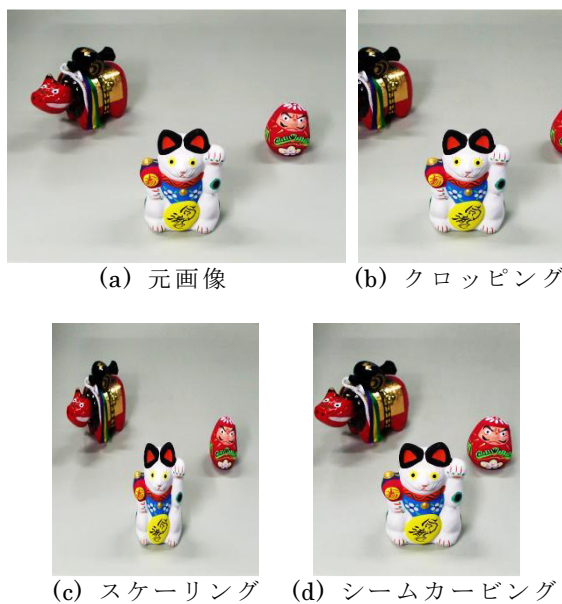


図 1 シームカービングによるリサイズの場合

シームカービングは、シーム (seam) と呼ばれる画素集合を削除することにより、画像のリサイズを行う。画像の幅を縮小する場合の例を図 2 に示す。図中の赤い線がシームである。シームを構成する画

素は 1 行に 1 個のみ存在し、シーム全体は連続な離散曲線となっている。画像を 2 ピクセル以上縮小する場合は、シームを求める処理を繰り返せばよい。

1 本のシームは、動的計画法により画素数に対する線形時間で求めることができるが、画像の大きさやリサイズの幅によっては、ソフトウェアによる処理では時間がかかっていた。画像のリサイズアルゴリズムは、実時間処理が必要な分野に応用されることが多いと見込まれる。そこで本研究では、シームカービングの処理時間短縮を目的として、アルゴリズムの処理を行う専用回路の設計を行った。



図 2 シーム

2. シームカービングアルゴリズム

シームカービングアルゴリズムでは、シームを求める際に、適当なエネルギー関数 $E(x, y)$ により各画素 $I(x, y)$ に固有のエネルギー値を割り当てる。エネルギー関数は、エッジ周辺で高い値をとるものであればよく、文献 [1] では式 (1) で与えられる勾配強度を表す関数を使用している。本研究でも、式 (1) をエネルギー関数として使用する。

$$E(x, y) = \left| \frac{\partial}{\partial x} I \right| + \left| \frac{\partial}{\partial y} I \right| \quad (1)$$

シームを構成する画素は、エネルギーの総和が低くなるように動的計画法により求める。 $M \times N$ 画素の画像を横方向に 1 ピクセル縮小する場合を例として、アルゴリズムの処理の過程を説明する。

- (1) 画素 $I(x, y)$ のエネルギー $E(x, y)$ を画素値とするエネルギーマップ E を作成する (図 3(a))。
- (2) シームの候補となる経路のエネルギー $\Sigma E(x, y)$ (最上段の行の画素から着目画素 $I(x, y)$ に至る経路上のエネルギーの総和) を画素値とするエネルギーマップ ΣE を作成する (図 3(b))。シームの候補の決定は、図 3(b) に示すように、着目画素の 1 行上の隣接 3 画素のうち、 $\Sigma E(x, y)$ が最小の画素を選択していくことで行う。
- (3) 図 4 に示すようなシーム画像を作成する。シーム画像中の画素値 (0, 1, 2) は、次に接続すべき画素の方向を示しており、それぞれ左上、真上、右上に対応する。この値をたどっていくことによって経路が定まる。
- (4) シーム画像が完成したら、画像の最下段の行上にある N 個の画素を起点とするシームの候補が N 本できる。これらの内で $\Sigma E(x, y)$ が最も小さく

FPGA implementation of seam carving algorithm

† Yasuhide KOHATA, Major in Media and Telecommunications Engineering, Graduate School of Science and Engineering, Ibaraki University

‡ Moemi KAZAMA, Yoshifumi YAGISAWA, Yoshihisa DESAKI, Department of Media and Telecommunications Engineering, Faculty of Engineering, Ibaraki University

なる画素を起点とするシームを削除する．シームを削除することによって横方向に1ピクセル縮小した画像ができる．

2ピクセル以上の幅で画像の縮小処理を行う場合は、以上の処理を繰り返せばよい．また、画像を縦方向に縮小する場合も、横方向に縮小する場合と同様の処理でリサイズすることが可能である．

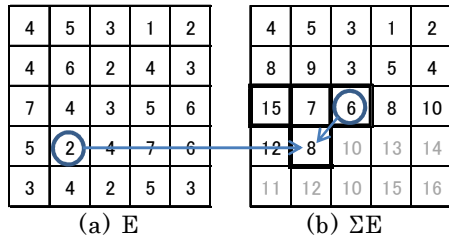


図3 エネルギーマップ

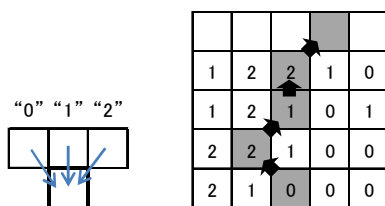


図4 シームの決定

3. シームカービング処理回路

本研究で設計したシームカービング処理回路のブロック図を図5に示す．

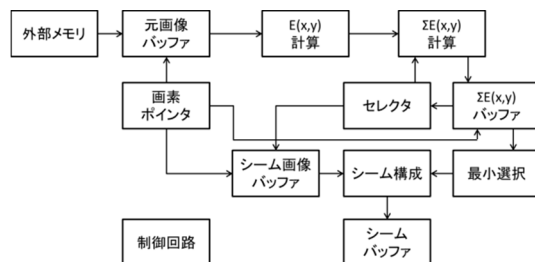


図5 シームカービング処理回路

設計した回路を構成する各ブロック、及び処理の流れについて説明する．

- 外部メモリ上の処理対象画像を元画像バッファに読み込む．
- $E(x,y)$ 計算回路で画素値の差分（勾配）をとり、画素ごとのエネルギー $E(x,y)$ を求める．
- $\Sigma E(x,y)$ 計算回路で、エネルギーの和 $\Sigma E(x,y)$ を求め、 $\Sigma E(x,y)$ バッファに記憶する．
- セクタでは、 $\Sigma E(x,y)$ の候補の中から最小のものを $\Sigma E(x,y)$ 計算回路に渡し、選択した $\Sigma E(x,y)$ の相対的な位置（図4左）をシーム画像バッファに記憶する．
- エネルギーの計算が全て終了した後、最小選択回路によって最終的な $\Sigma E(x,y)$ の候補の中からエネルギーが最小のものを選ぶ．
- シーム構成回路では、最小選択回路から渡された $\Sigma E(x,y)$ の座標を始点としてシーム画像バッファ内のシーム画像をたどってゆき、シームを決定して出力する．

- 画素ポインタは各ブロックに画素の座標を指定してその座標の値を読み出す．
- 制御回路により、回路全体の動作を制御する．

4. 性能評価

シームカービングアルゴリズムを、今回設計した専用回路、及びソフトウェアで実装した場合の性能評価を行う．エネルギー計算の処理に要する時間を評価指標とし、それぞれの実装方式の処理性能を比較する．各実装方式の開発環境を以下に示す．

[開発環境]

[専用回路]

HDL : Verilog HDL

開発環境 : ISE Design Suite 12.2 (Xilinx 社)

FPGA : VIRTEX-5, XC5VLX30 (Xilinx 社)

[ソフトウェア]

CPU : Intel Xeon W3565 (3.2GHz)

メモリ : 8GB

OS : Windows7 (64bit)

開発環境 : Microsoft Visual C++ 2010 Express

画像処理ライブラリ : OpenCV 2.4.2

640×480ピクセルの画像を処理対象とした場合のエネルギーマップの処理時間を表1に示す．専用回路の処理時間は、シミュレーションにより求めたアルゴリズムの処理に要する消費クロック数と、ターゲットデバイスとなるFPGAのグローバルクロック周波数（24MHz）から求めた値である．

各実装方式の処理時間を比較すると、専用回路による処理はソフトウェアによる処理に比べて約900倍高速になっている．ソフトウェアによる処理では、エネルギー計算を1ピクセルごとに行う必要がある．これに対して、今回設計した専用回路では、画像の1行分（640ピクセル）の処理を並列に行っているため、処理時間を大幅に短縮することができた．

表1 エネルギーマップの処理時間

	処理時間[μs]
専用回路	約 20
ソフトウェア実装	約 18,000

5. まとめ

本研究では、シームカービングアルゴリズムを処理する専用回路を設計し、その性能を評価した．リサイズの比率が50%～150%程度の場合、画像のサイズによっては実時間処理可能であることを示すことができた．

動画像をリサイズする場合はグラフカットアルゴリズム等の大域最適化手法が必要になるため、オンライン処理を行うためには工夫が必要である．また、フレームの集合に対する演算が必要になることから、メモリ消費量も増大する．これらの問題を解決するための実装方法を探ることを今後の課題とする．

参考文献

- [1] S. Avidan and A. Shamir, “Seam Carving for Content-Aware Image Resizing,” SIGGRAPH, 2007.