

汎用エンジン RM-IV のブラインド・デコンボリューションへの応用

2 F - 5

奥田 知史, 井上 真一, 沼 昌宏, 平野 浩太郎

神戸大学大学院 自然科学研究科

1. はじめに

ブラインド・デコンボリューションを用いた画像の復元における一手法として、シミュレーテッド・アニーリングを用いた手法 [1] がある。この手法では、ブラインド・デコンボリューションを最適化問題に置き換えている。二つの画像 $f(x, y)$, $g(x, y)$ のコンボリューション画像と解析すべき画像 $h(x, y)$ との差のエネルギー

$$Q(f, g, h) = En\{f(x, y) * g(x, y) - h(x, y)\} \quad (1)$$

$En\{\cdot\}$: エネルギーを表す演算子

*: コンボリューションを表す演算子

を評価関数として定め、 $f(x, y)$, $g(x, y)$ を変化させたときに Q が最小になる f と g の組を最適解、すなわち復元画像としている。しかしこの手法では反復処理を用いているため、復元画像が収束するまでに膨大な計算時間を必要とする。離散階調法を用いた高速化手法 [2] も提案されたが、十分ではなかった。そこで本稿では、汎用エンジン RM-IV [3] を用いて高速化を図る BDE (Blind Deconvolution Engine) について述べる。

2. ブラインド・デコンボリューション・エンジン : BDE

2.1 ハードウェア化のためのアルゴリズム変更

BDE では高速化のために、パイプライン処理に適したアルゴリズムに変更する必要がある。従来の手法 [2] では、コンボリューション画像の最大画素値 h'_{max} をもとに正規化係数 α'_n を

$$\alpha'_n = 255/h'_{max} \quad (2)$$

の式にしたがって導き、正規化が行われていた。

この場合、全画素のコンボリューション演算が終了するまで正規化係数 α'_n が得られないので、

パイプライン処理には適さない。この問題を解決するために、点像分布画像 $g(x, y)$ の各画素値の総和から正規化係数 α_n を導き、正規化を行うことにする。階調数を Ng (2~256) とすると、 α_n は

$$\alpha_n = \frac{1}{\sum_{x,y} g(x, y)} \times \frac{255}{Ng-1} \quad (3)$$

のように表される。この式を用いると、正規化係数 α_n はコンボリューション演算を行う前の段階で判明しているため、画像のエネルギー計算が一連の処理として行うことが可能となり、パイプラインに適したアルゴリズムとなる。またコンボリューション画像の最大値を探索する手間も省ける。

2.2 高速化のためのアルゴリズム変更

次に、エネルギー計算回数の削減による高速化を考える。従来は、毎回すべての画素についてエネルギー計算をしていたが、正規化係数が前回と同等のものならば、変化した部分はサポート部分、すなわち f または g の一画素の変化にともなって、 $f * g$ の画素が変化する部分 (図1の①) のみなので、全体のエネルギーは次のようになる。

$$\begin{aligned} & (\text{全体のエネルギー}) = (\text{前回の全体のエネルギー}) \\ & - (\text{変化前のサポート部分のエネルギー}) \\ & + (\text{変化後のサポート部分のエネルギー}) \quad (4) \end{aligned}$$

画像 g が変化しないときは正規化係数 α_n も変化しないので、前回の全体のエネルギー、更新前の各画素のエネルギーを保存することで、このときのエネルギー計算回数は 1/4 となる。

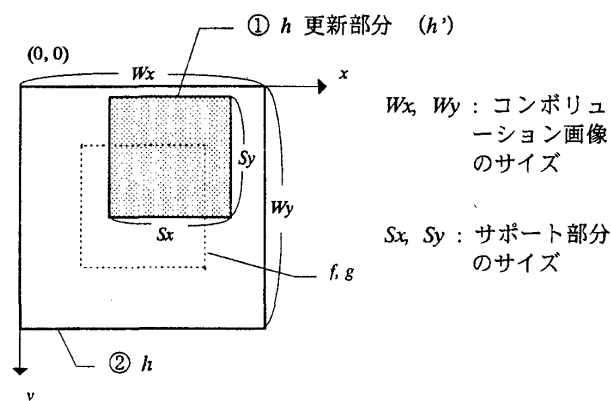


図1 画像モデル

Blind Deconvolution on Reconfigurable Machine-IV
Tomofumi Okuda, Shin-ichi Inoue,
Masahiro Numa, and Kotaro Hirano
The Graduate School of Science and Technology,
Kobe University

表1 ソフトウェアとの処理時間の比較

項目	BDE (4 MHz)	ソフトウェア
画像データ作成時間	0.66	—
データ転送時間	0.60	—
処理時間	7.90	143.00
データ転送時間	0.60	—
画像データ変換時間	0.05	—
総処理時間	9.81	143.00

単位 (s)

また従来の手法では、受理・棄却判定で受理された場合、すべてをメモリに記録していたが、変化した部分はサポート部分のみであるので、前回の画像にサポート部分のみを変更すればよい。これにより更新する画素の範囲が1/4になる。

2.3 パイプライン処理

エネルギー計算処理を細分化し、パイプライン処理によって高速化を図る。エネルギー計算処理を、一つの状態に要するクロック数が1になるような同時に実行可能な L 個の状態 (パイプライン段数) に分けたとき、処理を行う画素数を N とすると、クロック数は $N+L-1$ となる。BDE のパイプラインは17段である。

3. 実験と考察

シミュレーションによって求めた BDE の処理時間とソフトウェアとの比較を表1に示す。ソフトウェアの処理時間測定に利用した計算機は SPARCstation 5 互換機 (約 72 MIPS)、対象とした画像は 32×32 画素、2階調である。なお、このシミュレーションでは FPGA、FPIC の遅延も考慮している。また現在、 4×4 画素の画像については、クロック 4 MHz での実機動作を確認している。

BDE の実験結果に対する考察を以下に述べる。表1より、BDE のハードウェア部の処理速度はソフトウェアの約 18 倍であることがわかる。またシステム全体としても、約 14.5 倍の処理速度であることがわかる。以下、高速化の要因について考察を加える。

(1) アルゴリズム変更による高速化

コンボリューション画像の画素数を $P_n (=W_x \cdot W_y)$ 、画素値変更回数を N_p とし、受理判定時に $1/2$ の割合で受理されたとし、受理の回数 $N_a = N_p/2$ とする。これより BDE において、データパスを通る画素数を次のように表すことができる。

$$\frac{1}{2}N_p \cdot \frac{1}{4}P_n + \frac{1}{2}N_p \cdot P_n + \frac{1}{2}N_p \cdot \frac{1}{4}P_n = \frac{3}{4}N_p \cdot P_n \quad (5)$$

また、2.2 節で述べたアルゴリズムの高速化を行わない場合、データパスを通る画素数を次のように表すことができる。

$$\frac{1}{2}N_p \cdot P_n + \frac{1}{2}N_p \cdot P_n + \frac{1}{2}N_p \cdot P_n = \frac{3}{2}N_p \cdot P_n \quad (6)$$

BDE ではその処理の 90% 以上がデータパス部での処理なので、式 (5)、(6) からアルゴリズムの変更による高速化効率は次のようになる。

$$\frac{3}{2}N_p \cdot P_n / \frac{3}{4}N_p \cdot P_n = 2 \quad (7)$$

この式から、アルゴリズムの変更により約 2 倍の処理速度が得られたことがわかる。

(2) パイプライン処理による高速化

パイプライン処理による高速化効率について考える。BDE の総クロック数は、

$$Clk_{sum} \cong N_p + (S_x \cdot S_y + L - 1)N_p / 2 + (W_x \cdot W_y + L - 1)N_p / 2 + (S_x \cdot S_y + L - 1)N_a + N_a \quad (8)$$

のようになる。パイプライン処理方式を採用していない BDE を考えると、クロック数は、次のようになる。

$$Clk'_{sum} \cong N_p + S_x \cdot S_y \cdot L \cdot N_p / 2 + W_x \cdot W_y \cdot L \cdot N_p / 2 + S_x \cdot S_y \cdot L \cdot N_a + N_a \quad (9)$$

ここで $N_a = N_p/2$ とすると、式 (8)、(9) より 32×32 画素、2階調の画像に対して、パイプライン処理によって約 15 倍の処理速度が得られることがわかる。

4. まとめ

画像復元手法の一つであるブラインド・デコンボリューションのアルゴリズムの主要な部分について、汎用エンジン RM-IV を用いてハードウェア化したブラインド・デコンボリューション・エンジンを開発した。

参考文献

- [1] B. C. McCallum, "Blind deconvolution by simulated annealing", Opt. Commun., vol. 75, pp. 552-558, 1990.
- [2] 田丸雅也, 野村孝徳, 平野浩太郎, "シミュレーテッド・アニーリングを用いたブラインド・デコンボリューションにおける収束性の改善", 光学連合シンポジウム '94 講演会予稿集, pp. 77-78, 1994.
- [3] 井上真一, 高瀬幹, 沼昌宏, 平野浩太郎, "汎用エンジン RM-IV の構成", 情報処理学会第 53 回全国大会, 2F-04, 1996.