

3次元 DCT を効率的に実行するアレイプロセッサの提案

生垣 佑樹[†] 宮崎 敏明[†] Stanislav G. Sedukhin[†]会津大学コンピュータ理工学部[†]

〒965-8580 福島県会津若松市一箕町鶴賀

1 はじめに

本稿では、単純な機構を持つプロセッシングエレメントを 3 次元トーラス状に接続し、データ同士の入れ替え作業を陽に行わず、効率的に 3 次元 DCT (離散コサイン変換) を実行するアレイプロセッサを提案する。本アレイプロセッサは、データサイズ $n \times n \times n$ の 3 次元 DCT を $3n$ ステップで実行できる。ただし、 n は 1 以上の整数。

2 提案アレイプロセッサ機構

図 1 に提案するアレイプロセッサのプロセッシングエレメント (以下、PE と略す) の構造を示す。PE は主に、入出力セレクタ、一組の積和演算器と、 $3n+4$ 個のレジスタで構成される。

図 2 に各 PE の、隣接 PE との接続と、データ転送方向を示す。破線の矢印は PE のトーラス接続を意味する。全ての PE は同じ構造をしており、演算処理は、データ転送と演算を交互に繰り返すシストリック動作を基本としている。処理全体の流れは、データ入力、3D-DCT 処理ステップ 1, 2, 3 と、データ出力からなる。

本機構では、 $n \times n$ 個のデータを同時に入力する。入力ポートは $k=n-1$ 平面上の PE に向けて配置する。各 PE が、 k 軸負方向の隣接 PE に自分が受け取った入力データを転送する処理を n ステップ繰り返すことで、全 PE に入力データが転送される。出力ポートは $k=0$ 平面上の PE からデータを受け取るよう配置される。入力と同様の転送処理を n ステップ繰り返すことで、出力は完了する。

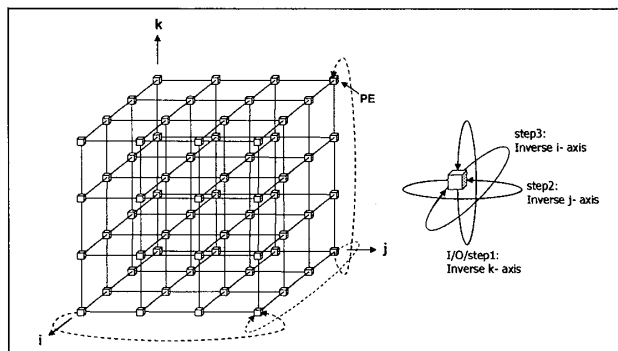


図 2 : PE の接続図およびデータ転送方向

$X_{n \times n \times n} = [X(i, j, k)]$, $0 \leq i, j, k \leq n-1$ について、3D-DCT の計算結果 $Y_{n \times n \times n} = [Y(s, r, p)]$, $0 \leq s, r, p \leq n-1$ は以下のよう定義される。

$$Y(s, r, p) = \Phi \cdot \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} X(i, j, k) \times C(k, p) \times C(j, r) \times C(i, s),$$

$$\Phi = \sqrt{\frac{8}{n}} \cdot \varepsilon(s) \cdot \varepsilon(r) \cdot \varepsilon(p),$$

$$\varepsilon(m) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } m = 0; \\ 1, & \text{for } m > 0; \end{cases} \quad m \in \{s, r, p\},$$

また、係数行列 $C_{n \times n} = [C(u, v)]$ は

$$C(u, 0) = \frac{1}{\sqrt{2}}, \quad C(u, v) = \cos\left(\frac{\pi(2u+1)v}{2n}\right),$$

$u \in \{i, j, k\}, (u, v) \in \{(i, s), (j, r), (k, p)\}$ と定義される。

図 3 : 3D-DCT

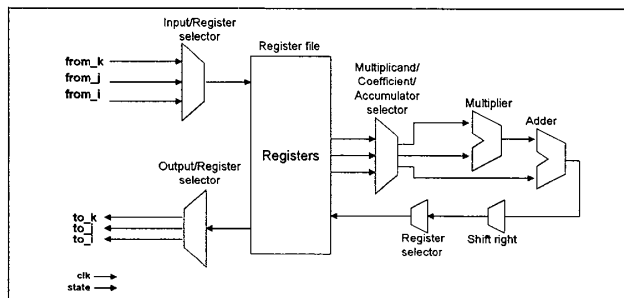


図 1 : PE の構造

図 3 に 3D-DCT の式を示す。本機構での 3D-DCT 処理ステップは 3 段階に分けられる。ステップ 1 では $X(i, j, k)$ と $C(k, p)$, ステップ 2 では $X(i, j, k)$ と $C(j, r)$, ステップ 3 では $X(i, j, k)$ と $C(i, s)$ の乗算と、結果の累積加算が行われる。各ステップは n 個の micro-step に分けられ、各 micro-step での PE の処理の流れは以下の通りで

ある：(1)隣接 PE からデータを受け取り、レジスタに格納する。(2)受け取ったデータと、予め PE のレジスタに格納しておいた係数を掛け合わせ、計算結果をレジスタに累積加算する。(3)最後に(1)で受け取ったデータを隣接 PE に転送する。ステップ 1, 2, 3 で転送されるデータはそれぞれ、入力データ、ステップ 1 の計算結果、ステップ 2 の計算結果である。PE (i, j, k) で、各 micro-step で計算に用いられる係数のインデックスは以下の通りである。

- ステップ 1: $C((k + \text{microstep}) \bmod n, k)$
- ステップ 2: $C((j + \text{microstep}) \bmod n, j)$
- ステップ 3: $C((i + \text{microstep}) \bmod n, i)$

3D-IDCT では係数 C の転置行列を計算に用いる。

n^3 -サイズの 3D-DCT 計算について、直接計算では n^3 の時間ステップ数を要するのに対し、提案機構では $3n$ で済む。

3 提案機構のパイプライン化

本章では、スループットの改善のため、前章で提案した機構のパイプライン化を行った。以下、2 種類の機構

An Array Processor Performing 3D-DCT Effectively
[†] Yuki Ikegaki, Toshiaki Miyazaki, Stanislav G. Sedukhin
 School of Computer Science and Engineering,
 The University of Aizu

について述べる。まず、1つめのパイプライン機構(以下、P1と略す)の動作概要を図4に示す。各欄の上段は、そのステージにおける処理、下段がその処理に用いるデータ転送路を表す。P1では、前章で提案した機構(以下、Seq)に対し、2対の積和演算器が必要となる。また、入力ポートを $i=n-1$ 平面上のPEに向けて配置し、入力データの転送方向は i 軸負方向となる。次に、2つめとなる、P1から更に高速化を図ったパイプライン機構(以下、P2と略す)の動作概要を図5に示す。図において、InputおよびOutput行は、対応するステージの入出力データがどのステージで用いられるデータであるかを表す。P2ではSeqに対してI/O専用線、3対の積和演算器および $3N+12$ 個のレジスタを必要とする。P2ではハードウェアコストが増し、配線がより複雑化することは明らかではあるが、理論上、Seqに比べてP1は2.5倍、P2は5.0倍、高速に動作する。

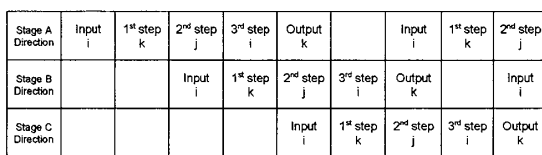


図4: Pipelined ver.1 パイプライン図

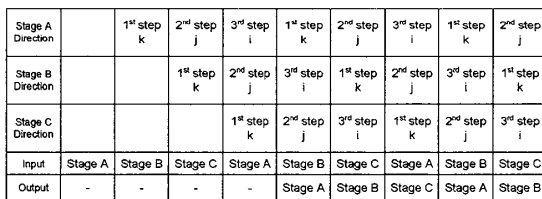


図5: Pipelined ver.2 パイプライン図

4 $(n-1)^3$ -サイズ アレイプロセッサを用いた n^3 -サイズ 3D-DCT 計算機構

本稿で提案するアレイプロセッサは $O(n^3)$ のハードウェアリソースを必要とするため、実装において、それらのコストを減少させることは重要である。本章では、 $(n-1)^3$ サイズのアレイプロセッサを用いて n^3 サイズの3D-DCT 計算結果を得る機構を提案する。図6に、 $n=8$ の場合の機構の全体図を示す。本機構の特徴は以下の通りである：

- 8x8x8-サイズ 3D-DCT の計算を8つの4x4x4-サイズに分割して計算する。
- i - j - k -軸すべてのトラス接続に、FIFOを挟み込む。各FIFOは4つのデータを保持する。
- 各4x4x4-サイズの計算結果を得るには3x8ステップ要する。
- 演算器数が、8x8x8-サイズの機構の1/8で済む。
- 各PEに予め保持させておく係数は8x8x8-サイズの機構のそれと同じである。
- 1つの4x4x4-サイズの計算には $4^3 \times 4$ 個のデータ入出力が必要となる。

本計算機構では、2、3章で提案した機構と比較して演算器数が指数関数的に減少するため、FPGAのような、面積の限られているハードウェアへの実装を容易にする。

5 性能評価

提案機構の効率性を評価するために、リアルタイム処

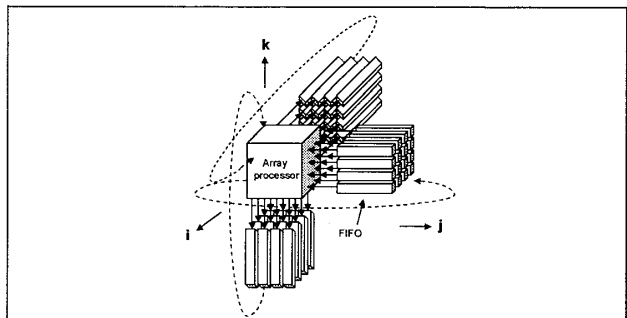


図6: 4x4x4-サイズ アレイプロセッサを用いた8x8x8-サイズ 3D-DCT 計算機構

理に要する理論上の最低動作周波数を求めた。 $n=8$ 、入力データは単色、1 micro-stepが2クロックで処理されるとした場合の結果を表1に示す。結果から、頻繁に利用されると考えられるVGAサイズの動画に対し、32, 60fps時ともに数100kHz未満の低動作周波数でのリアルタイム動作が得られ、低消費電力に貢献すると考えられる。また、UXGAサイズ/60fps時の大画面/高画質の場合についても数MHzの動作周波数が得られ、来たる携帯機器でのHD動画再生の需要に応えるものと考えられる。RGBカラーの動画に対しては表内の値の3倍、4章で提案した機構を利用する場合は8倍、最低動作周波数は大きくなる。

表1: リアルタイム処理に要する、Seq, P1, P2各機構での最低動作周波数 ($n=8$)

fps = 32	Size	Min. freq. for real-time en/decoding (Seq)	Min. freq. for real-time en/decoding (P1)	Min. freq. for real-time en/decoding (P2)
QVGA	320x240	48.0 [KHz]	19.3 [KHz]	9.7 [KHz]
VGA	640x480	192.0 [KHz]	76.9 [KHz]	38.5 [KHz]
SVGA	800x600	300.0 [KHz]	120.1 [KHz]	60.1 [KHz]
XGA	1024x768	491.6 [KHz]	196.7 [KHz]	98.4 [KHz]
SXGA	1280x1024	819.2 [KHz]	327.7 [KHz]	163.9 [KHz]
UXGA	1600x1200	1,200.0 [MHz]	480.1 [KHz]	240.1 [KHz]
fps = 60	Size	Min. freq. for real-time en/decoding (Seq)	Min. freq. for real-time en/decoding (P1)	Min. freq. for real-time en/decoding (P2)
QVGA	320x240	90.0 [KHz]	36.1 [KHz]	18.1 [KHz]
VGA	640x480	360.0 [KHz]	144.1 [KHz]	72.1 [KHz]
SVGA	800x600	562.5 [KHz]	225.1 [KHz]	112.6 [KHz]
XGA	1024x768	921.6 [KHz]	368.7 [KHz]	184.4 [KHz]
SXGA	1280x1024	1,536.0 [KHz]	614.5 [KHz]	307.3 [KHz]
UXGA	1600x1200	2,250.0 [KHz]	900.1 [KHz]	450.1 [KHz]

6 おわりに

データサイズ $n \times n \times n$ の3次元DCTを $3n$ ステップで実行できるアレイプロセッサ、そのスループットおよびハードウェアコストを改善した機構を提案した。また、リアルタイム処理に要する最低動作周波数を求め、提案アーキテクチャの高効率実行性を示した。今後は入出力機構を含めた実機による評価を行い、本機構の実用性を検証する。

参考文献

[1] O. Alshibami, and S. Boussakta, "Fast Algorithm for the 3-D DCT," Proc of the IEEE International Conference on Acoustics, Speech, and Signal Processing 2001, pp. 1945-1948, 2001.

[2] M. Modarressi, and H. Sarbazi-Azad, "Parallel 3-Dimensional DCT Computation on k-Ary n-Cubes," Proc. of the 8th International Conference on High Performance Computing in Asia Pacific Region (HPC Asia 2005), pp. 91-97, 2005.