

エフェクト付き動画のためのマルチプロセッサシステムの提案

野津 隆弘[†] 吉村 龍洋^{††} 坂主 圭史[†] 武内 良典[†] 今井 正治[†]

[†] 大阪大学 大学院情報科学研究科 情報システム工学専攻

〒 565-0871 大阪府吹田市山田丘 1-5

^{††} 株式会社アクセル

〒 101-8973 東京都千代田区外神田 4-14-1 秋葉原 UDX 南ウイング 10 階

E-mail: †{t-notu,sakanusi,takeuchi,imai}@ist.osaka-u.ac.jp, ††yoshimura@axell.co.jp

あらし 近年、組み込み機器でエフェクト付き動画が用いられるようになってきているが、画像エフェクト処理の一種である座標変換エフェクトの処理は複雑な数学関数の計算が多い。そのため、座標変換エフェクトを用いたエフェクト付き動画は小さな画像にしか適用できなかった。本研究では、システムの演算処理性能を向上させ、より大きな画像に対してもなめらかな動画の生成を可能とするため、画像処理用プロセッサを二つ用いたマルチプロセッサ型画像エフェクト処理システムを提案する。提案システムでは、画像処理用プロセッサを単純に二つ用意するのではなく、一部の演算器を両方のプロセッサで共有することで、面積の増加を抑えている。マルチプロセッサ化した提案システムは、VGA 画像に施す 6 種のエフェクト生成速度をなめらかな動画を表示するために必要な、秒間 30 枚という目標を達成している。

キーワード 画像エフェクト処理 マルチプロセッサ ASIP 数学関数 並列化

A Proposal of Multiprocessor System for Effected Movies

Takahiro NOTSU[†], Tatsuhiro YOSHIMURA^{††}, Keishi SAKANUSHI[†], Yoshinori TAKEUCHI[†],
and Masaharu IMAI[†]

[†] Graduate School of Information Science and Technology, Osaka University

1-5 Yamada-Oka, Suita-shi, Osaka, 565-0871, Japan

^{††} AXELL Corporation

Akihabara UDX SouthWing 10F, 4-14-1 Sotokanda, Chiyoda-ku, Tokyo, 101-8973, Japan

E-mail: †{t-notu,sakanusi,takeuchi,imai}@ist.osaka-u.ac.jp, ††yoshimura@axell.co.jp

Abstract Recently effected movies have been used in embedded systems. Image effect processing with coordinate transformations needs complicated calculations. Only small images can be generated when coordinate transformation effect was used on the movie. In this report we propose a new effect processing system with two processors, increasing arithmetic performance in order to make smooth effected movies and try to avoid increasing the area of the system by means of shareing some computing units. The proposed system can process effects on VGA size images at the rate of 30 frame per second and can generate smooth effected movies.

Key words Image effect processing, Multiprocessor, ASIP, Mathematical function, parallelization

1. はじめに

近年、携帯用組み込みシステムでは、使用者の利便性を高めるための GUI や、製品の付加価値を向上させるという目的のために高い解像度で精細なグラフィックアニメーションが使用されるようになってきている。そのようなグラフィックアニメーションの一つにエフェクト付き動画がある。エフェクト付き動

画とは入力動画画像に対し連続的にエフェクトを施したり、静止画像に対しエフェクトのパラメータを時間とともに変化させることで生成された動画である。

1 枚の画像に対する画像エフェクト処理の処理内容は複雑であり、対象となる画像が大きくなるに従って計算量も増大する。視覚的になめらかな動画を得るためには 1 秒間に 30 枚程度での画像生成能力が必要となるが、組み込み機器に実装されて

いる汎用プロセッサは計算能力が低いいため大きな画像に対しエフェクト付き動画を生成することは難しい。したがって、携帯用組み込みシステムで大きな画像のエフェクト付き動画を使用するためには画像エフェクト処理に特化したシステム、画像エフェクト処理システム、が必要になる。

一般に、規模の大きなシステムは処理能力も高いが、携帯用組み込みシステムでは要求される設計制約が厳しいためシステムの規模を抑える必要がある。このような設計制約の元で、プロセッサの性能を向上させる方法としては、VLIW(Very Long Instruction Word)化やマルチプロセッサ化によりプロセッサの命令実行の並列度を高める方法や、特定の処理を専用演算器で実装しプロセッサの拡張命令でその演算器を使用する、といった方法がある。文献[2]の画像処理システムでは、動き予測や最大値最小値の計算といったの低レベルの画像処理にはパイプラインの深い演算器を用いて、VLIW命令で制御することにより性能の向上と柔軟性を両立させている。一方、特定の処理内容を一つの専用演算器に置き換え、それをプロセッサの拡張命令を用いて使用する方法としては、吉村らが提案した画像エフェクト処理用のシステムがある[1]。このシステムはピクセルに対する処理や数学関数の演算に特化した演算器や、画像エフェクト処理に特化した転送システムを実装している。QVGAサイズ(320×240)の画像に対しては様々な画像エフェクト処理を1秒あたり30枚の画像に対し実行することができ、この大きさのなめらかな動画の表示が可能である。しかし、画像の大きさがVGAサイズ(640×480)になると座標変換エフェクトの処理内容が増大し、目標とする時間内で画像エフェクト処理を行うことが難しくなる。したがって、座標変換エフェクトの処理能力を向上し、より大きな画像に対してもなめらかなエフェクト付き動画を生成できるシステムを開発することが課題となっている。

そこで本研究では[1]のシステムに実装されているプロセッサをマルチプロセッサ化して命令実行の並列度を上げることで、VGAサイズの画像に対して1秒間に30枚の画像に対しエフェクト処理を実行し、なめらかなエフェクト付き動画を生成できるような画像エフェクト処理システムを提案する。また、使用頻度が低い、あるいは、パイプライン化されており複数の演算を並列して実行できる演算器をプロセッサ間で共有することによりシステムの規模の増大を抑えることを提案する。なお、本研究ではプロセッサの並列化による演算能力の向上について着目するため、画像メモリのアクセス方法については考慮しない。以下、本稿の構成を示す。第2節で画像エフェクト処理全般について簡単に説明し、第3節で吉村らが提案している画像エフェクト処理システムについて説明する。第4節で提案するマルチプロセッサ型画像エフェクト処理システムについて説明し、第5節で提案したシステムの評価を行う。最後に第6節で総括と今後の課題を述べる。

2. 画像エフェクト処理

画像エフェクト処理とは与えられた画像に対し、視覚的な変形を施す処理である。この処理はPhotoshopやGIMPなどの

コンピューターグラフィックスソフトウェアで、フィルタとして知られている。画像エフェクト処理はピクセル変換エフェクトと座標変換エフェクトの二種類に分類することができる。

ピクセル変換エフェクトとは、入力画像のピクセル値を変換して出力画像を生成する処理である。ピクセル値の変換とは、ピクセルの全てもしくは一部の成分に対して加減乗算や論理演算を行うことである。

一方、座標変換エフェクトとは出力画像のそれぞれのピクセルの座標 (x_o, y_o) に対応する入力座標 (x_i, y_i) を算出し、その入力座標のピクセル値を出力画像の領域に転送することで出力が像を生成する処理である。座標の変換には四則演算や三角関数演算、距離および偏角演算などが含まれる。

3. 画像エフェクト処理システム

本節では吉村らが提案した画像エフェクト処理システム[1]について説明する。

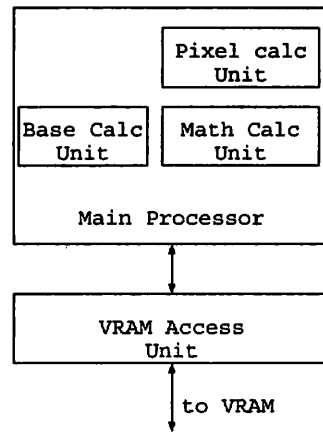


図1 画像エフェクト処理プロセッサのアーキテクチャ

図1に吉村らが提案した画像エフェクト処理システム[1]の構成を示す。このシステムは、画像エフェクト処理を行うプロセッサ(Main Processor)と画像メモリ(VRAM)アクセスユニット(VRAM Access Unit)から構成されており、プロセッサ内部には画像データのピクセル値に対する演算を行うピクセル演算ユニット(Pixel Calc Unit)と、三角関数や距離および偏角を計算する数学関数演算ユニット(Math Calc Unit)を持つ。プロセッサはDLXプロセッサ[4]の命令セットとピクセル演算ユニットと数学関数演算ユニットを使用する専用命令をもつ特定用途向け命令セットプロセッサ(ASIP)として実装している。これらの演算ユニットを専用命令から用いることでピクセル変換エフェクトおよび座標変換エフェクトを効率よく処理することができる。また、VRAMアクセスユニットは画像エフェクト処理に特化したメモリアクセス、書き込みアドレスの自動生成、先読みなど、を行うことによりVRAMアクセスに関するオーバーヘッドを軽減している。

このシステムではプロセッサを一つしか実装していないため、

一つ一つの処理を逐次的にしか実行できない。そのため、専用演算器を用いることによりある程度の高速化を実現することは可能であるが、座標変換エフェクトの処理のように複雑な演算を伴うエフェクトは、画像サイズの拡大とともに十分な画像生成速度を達成することが難しくなる。

4. マルチプロセッサ型画像エフェクト処理システム

本節ではマルチプロセッサ型画像エフェクト処理システムを提案する。まずシステムのアーキテクチャについて述べ、次にプロセッサの命令セットについて述べる。最後に並列化したプロセッサ間でのデータ通信について述べる。

4.1 アーキテクチャ

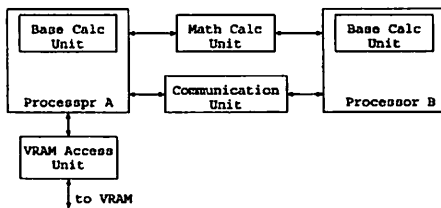


図2 提案システムのアーキテクチャ

図2に提案システムのアーキテクチャの構成を示す。システムは、二つのプロセッサ (Processor A, Processor B)、数学関数演算ユニット (Math Calc Unit)、コミュニケーションユニット (Communication Unit)、画像メモリ (VRAM) アクセスユニット (VRAM Access Unit) から構成され、それぞれのプロセッサの内部には基本演算ユニット (Base Calc Unit) がある。

VRAM アクセスユニットはプロセッサ A のみで使用でき、数学関数演算ユニットとコミュニケーションユニットはプロセッサ A, B の両方で使用できる。

4.1.1 プロセッサ

提案システムは二つのプロセッサを内蔵している。それぞれのプロセッサは基本演算ユニットを持ち、加減算、シフト演算、論理演算、分岐処理などを行う。ただし、乗算および除算は基本演算ユニットではなく共有されている数学関数演算ユニットを用いることで実現する。

4.1.2 数学関数演算ユニット

数学関数演算ユニットは乗算演算、除算演算、三角関数演算、CORDIC [3] 演算を行い、両方のプロセッサから使用できる。

数学関数演算ユニットは、複数のプロセッサで共有されているため、同じ演算を同時に実行することはできない。つまり、一つのプロセッサがある演算を実行しているときは、他のプロセッサはその演算が終了するまで待つことになる。ただし、乗算演算はパイプライン化されているため、別のプロセッサにより使用中であっても、演算の終了を待つことなく、1サイクル後には演算を開始することができる。

4.1.3 VRAM アクセスユニット

VRAM アクセスユニットはプロセッサと画像メモリ (VRAM)

との間に置かれ、画像エフェクト処理に適したメモリアccess機能を提供する。

しかし、本研究では座標変換エフェクト処理の高速化を対象とし、メモリアccessについては考慮していないため、座標変換エフェクト処理で必要となるピクセル値自動転送機能のインターフェースのみを実装している。

ピクセル値自動転送機能とは、プロセッサから出力画像上の座標と入力画像上の座標を受け取り、入力画像上の指定された座標のピクセル値を出力画像上の指定された座標に転送する機能である。VRAM アクセスユニットによるピクセル値自動転送機能については [1] を参照されたい。

4.1.4 コミュニケーションユニット

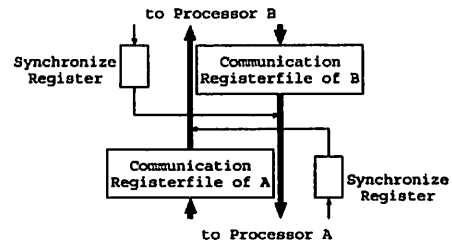


図3 コミュニケーションユニット

コミュニケーションユニットは、二つのプロセッサ間のデータ通信と同期処理を行う。図3にコミュニケーションユニットのアーキテクチャ構成を示す。コミュニケーションユニットは、個々のプロセッサに属する通信用レジスタファイルおよび同期レジスタから構成される。プロセッサは自身に属している通信用レジスタファイルのみに書き込むことができる。他のプロセッサに属する通信用レジスタファイルに書き込むことはできず、そのレジスタファイルの読み込みは可能である。

同期処理は同期レジスタを用いて実装する。同期レジスタは“SIGNALLED”か“NOT-SIGNALLED”の二つの状態をとる。プロセッサは読み込み先の同期レジスタの状態が“SIGNALLED”であるとき、他のプロセッサの通信用レジスタファイルの内容を読み込むことができる。同期レジスタの状態が“NOT-SIGNALLED”であれば、読み込みはブロックされる。

通信用レジスタの読み書き、および同期レジスタの制御は、次に示すプロセッサ同期命令を用いて行なう。

- GETCREG
- SETCREG
- CSIGNAL

GETCREG 命令は通信用レジスタファイルからデータを読み込む命令、SETCREG 命令は通信用レジスタファイルにデータを書き込む命令、そして、CSIGNAL は同期レジスタを制御する命令である。

4.2 命令セット

本節では提案システムのプロセッサの命令セットを示す。命令セットは DLX プロセッサ [4] を基としており、命令の種類は次の4種類に大別される。

- 基本命令
- 数学関数命令
- VRAM アクセス命令
- プロセッサ間の同期命令

以下、それぞれの命令の種類について詳しく述べる。

4.2.1 基本命令

基本命令には、加減算演算、シフト演算、比較演算、無条件および条件付分岐を行う命令が含まれる。これらの命令は DLX の整数演算命令にほぼ一致する。実装には、プロセッサ内部の基本演算ユニットと数学関数演算ユニットを用いる。

4.2.2 数学関数命令

数学関数命令には、乗算演算、除算演算、三角関数演算、距離および偏角演算を行う命令が含まれる。これらの演算命令はプロセッサ間で共有されている数学関数演算ユニットを用いる。

4.2.3 VRAM アクセス命令

VRAM アクセス命令には、VRAM アクセスユニットのピクセル値自動転送機能に必要な入力画像上の座標と出力画像上の座標を指定する命令がある。

4.2.4 プロセッサ同期命令

プロセッサ同期命令には、通信用レジスタファイルの書き込みを行なう SETCREG 命令、ブロッキングを伴う読み込みを行なう GETCREG 命令、そして同期レジスタの制御を行なう CSIGNAL 命令が含まれる。

4.3 プロセッサ間のデータ通信

本節ではコミュニケーションユニットを用いたプロセッサ間のデータ通信について説明する。

図 4 に並列化の対象となるプログラムを示す。このプログラムは二つのサブプログラム subProcedureA, subProcedureB からなり、それぞれ独立して実行が可能である。

システムをマルチプロセッサ化した場合、subProcedureB をプロセッサ B で実行し、それ以外の処理をプロセッサ A で実行することになる。実行の様子を図 5 に示す

```

1: procedure(a, b)
2: {
3:   c = subProcedureA(a);
4:   d = subProcedureB(b);
5: }

```

図 4 並列化の対象となるプログラム

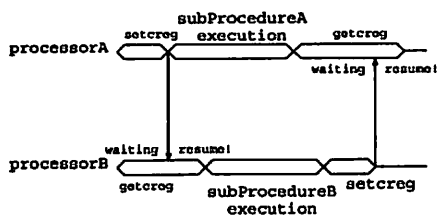


図 5 プロセッサ間のデータ通信の様子

別プロセッサでサブプログラム subProcedureB を実行する

ためには、まず引数 b をプロセッサ B に転送する必要がある。プロセッサ間のデータ転送には、通信用レジスタファイルを使用する。図 6 にプロセッサ A が実行するプログラムを示す。このプログラムは始めに、SETCREG 命令を用いて引数 b を通信用レジスタファイルに書き込む。次に、CSIGNAL 命令を用いて同期レジスタを設定し、プロセッサ B に対し引数 b が使用可能であることを通知する。

プロセッサ B が実行するプログラムを図 7 に示す。プロセッサ B は始めに GETCREG 命令を用いて引数 b を読み込む。引数の読み込みはプロセッサ A が CSIGNAL 命令を実行した後に完了する。なぜなら、プロセッサ A が CSIGNAL 命令を実行するまでは同期レジスタが設定されず、プロセッサ B の GETCREG 命令の実行がブロックされるためである。

プロセッサ A と B とでデータの通信が完了するとそれぞれのサブプログラムを実行する。プロセッサ B はサブプログラムの実行が完了すると、その実行結果 d を SETCREG 命令で通信用レジスタファイルに書き込み、CSIGNAL 命令を用いて同期レジスタを設定する。プロセッサ A は GETCREG 命令を用いて実行結果 d を読み込み、並列実行を終了する。

```

1: procedure(a, b)
2: {
3:   setcreg(b);
4:   csignal();
5:
6:   c = subProcedureA(a);
7:   d = getcreg();
8: }

```

図 6 プロセッサ A が実行するプログラム

```

1: procedure()
2: {
3:   getcreg(b);
4:   d = subProcedureB(b);
5:
6:   setcreg(d);
7:   csignal();
8: }

```

図 7 プロセッサ B が実行するプログラム

プロセッサ間のデータ通信ではしばしば待ち時間が発生する。これらの時間は「同期待ち時間」と呼ばれ、その間プロセッサの動作は停止する。この待ち時間が多いと演算処理が行われず、マルチプロセッサの性能を十分に発揮することができない。

5. 提案システムの評価

本節では、第 4 節で提案したシステムの性能の計測と回路規模の評価を行なう。

5.1 提案システムの性能

最初にシステムの性能評価の前堤となるシミュレーション環境について述べ、次にシステムの性能の計測結果を示す。

システムの性能は Mentor Graphics 社の ModelSim を用いて RTL シミュレーションを行い、画像エフェクト処理の実行に要するサイクル数から計測し、エフェクト画像の秒間生成枚数 (FPS)、高速化率、稼働率を算出する。

ここで、高速化率 (SpeedRate) とはマルチプロセッサ化した提案システムが吉村らの従来のシングルプロセッサ構成のシステム [1] と比較してどれだけ速く画像エフェクト処理を行うことができるのかという指標である。この指標は次に示す式で求める。

$$SpeedRate = \frac{CyclesOfSingleProcessor}{CyclesOfMultiProcessor}$$

$CyclesOfSingleProcessor$ はシングルプロセッサシステムでのエフェクトの実行サイクル数であり、 $CyclesOfMultiProcessor$ はマルチプロセッサシステムでのエフェクトの実行サイクル数である。

また、稼働率 (OperationRate) とは、画像エフェクト処理の実行時間のうち、実質的な演算、つまり、同期待ち以外の演算を行った割合を示す。この値は次に示す式で求められる。

$$OperationRate = 1 - \frac{ProcessorAWait + ProcessorBWait}{CyclesOfMultiProcessor \times 2}$$

$ProcessorAWait$ は ProcessorA が同期待ちを行うサイクル数の合計であり、 $ProcessorBWait$ は ProcessorB が同期待ちを行うサイクル数の合計である。これらのサイクル数は命令セットシミュレータ (ISS) を用いて測定する。

表 1 シミュレーションや評価に用いたパラメータを示す。実際のシステムでは画像メモリ (VRAM) とプロセッサとの間の画像データの転送に要するオーバーヘッドが存在する。しかし、この実験ではマルチプロセッサシステムの画像エフェクト処理に対する演算能力を確認することが目的であるため、VRAM 転送のオーバーヘッドが存在しない理想的な環境を想定している。

表 1 シミュレーション環境

動作周波数	167MHz
画像サイズ	640 × 480 [pix × pix]
VRAM 転送のオーバーヘッド	なし

計測対象のエフェクトは次の 6 種類とした。

- うずまき
- 波紋
- ジグザグ
- 球面
- 直交座標→極座標
- 極座標→直交座標

表 2 に、エフェクト処理をシングルプロセッサ上で実行させたときとマルチプロセッサ上で実行させたときでの実行サイクル数を示す。表より、提案するマルチプロセッサ型画像エフェクト処理システムは、測定に用いた全ての画像エフェクト処理

で画像生成枚数が目標の 30 を超えていることが分かる。

表 3 にそれぞれの画像エフェクト処理の高速化率と稼働率を示す。高速化率は 1.32 から 1.84 倍まで、となっている。もともと実行サイクル数が少ないエフェクトについては高速化しにくい、実行サイクル数が大きいものについては高い高速化率が得られた。

稼働率は 0.80 から 0.97 までとなっており、概ね高い。これは、座標変換エフェクトでは一つ一つの座標の変換がそれぞれ独立しており、ループ処理のソフトウェアパイプライン化によりタスクの分割が容易に行えるためである。また、稼働率が高いと高速化率も高くなるが、プロセッサ間のデータ転送が多いと高速化率は低くなる。

表 2 画像エフェクト処理の実効速度の比較

エフェクト名	秒間生成枚数	
	従来システム	提案システム
うずまき	28.12	40.05
波紋	22.17	36.44
ジグザグ	24.32	44.85
球面	23.9	34.11
直交座標→極座標	32.86	45.84
極座標→直交座標	44.25	58.19

表 3 高速化率と稼働率

エフェクト名	高速化率	稼働率
うずまき	1.42	0.92
波紋	1.64	0.97
ジグザグ	1.84	0.94
球面	1.43	0.80
直交座標→極座標	1.40	0.84
極座標→直交座標	1.32	0.89

5.2 提案システムの回路規模

提案システムを論理合成し、回路規模を測定した。

論理合成には Synopsys 社の DesignCompiler を使用し、0.14μm の CMOS ライブラリを用いた。面積制約は与えず、クロック制約を 200MHz として合成した。

論理合成の対象はプロセッサ単体、数学関数演算ユニット、システム全体とした。プロセッサ単体とは、図 2 のプロセッサ A (Processor A) のみの回路であり、システム全体とは図 2 の内 VRAM アクセスユニット (VRAM Access Unit) を除いた回路である。

表 4 論理合成結果

測定対象	面積 (ゲート換算)
プロセッサ単体	41382.0
数学関数演算ユニット	27365.7
システム全体	120186.9

表 4 論理合成の結果得られた面積を示す。表 4 よりマルチプロセッサ型画像エフェクト処理システム全体で 120,000 ゲートの大きさであった。また、共有されている数学関数演算ユニッ

トの面積はおよそ 27,000 ゲートであり、数学関数演算ユニットも並列化した場合に比べてこれだけの面積が削減できていることになる。

今回の実験では、ピクセル演算器と VRAM アクセスユニットは実装していないが、ピクセル演算器と VRAM アクセスユニットの面積は [1] での測定結果よりそれぞれ 50,000 ゲート、35,600 ゲートである。もし、現在のシステムに VRAM アクセスユニットとピクセル演算器を含めれば、205,786 ゲートの規模になると予想される。

6. まとめと今後の課題

本稿では、処理に時間を要する座標変換エフェクトの処理をマルチプロセッサ化により高速化を試みた。その結果、画像エフェクト処理の秒間実行枚数が目標の 30 を超えたエフェクトは、シングルプロセッサ環境では 6 種類中 2 種であったが、マルチプロセッサ環境では測定に用いた全てのエフェクトで秒間実行枚数が 30 を超えることができた。この結果より、座標変換エフェクトに関しては提案したマルチプロセッサシステム有効であると言える。

本研究で取り扱ったエフェクトは座標変換エフェクトのみでありピクセル変換エフェクトについては別途評価する必要がある。また、本研究では演算性能のみに着目しているが画像メモリとプロセッサとの間のピクセルデータの通信のコストも考慮する必要がある。今後の研究では、ピクセル変換エフェクトも含めた様々なエフェクトや画像メモリとプロセッサとの通信を考慮した評価を行う。

文 献

- [1] 吉村 龍洋, 坂主 圭史, 武内 良典, 今井 正治, "実時間画像エフェクト処理のための ASIP アーキテクチャの提案," 電子情報通信学会技術研究報告, VLD2004-118, Vol. 104, No. 590, pp.49-54, 2005 年
- [2] W. Kruijtzter, W. Gehrke, V. Reyes, G. Alkadi, T. Hinz, J. Jachalsky and B. Steux, "The Design of a Smart Imaging Core for Automotive and Consumer Applications: A Case Study," Proceedings of CODES+ISSS 2005, pp. 124-129, Sept. 2005.
- [3] R. Andraka and Andraka Consulting Group, Inc., "A survey of CORDIC algorithms for FPGA based computers," Proceedings of International Symposium on Field Programmable Gate Arrays 98, pp. 191-200, Feb. 1998.
- [4] John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann Publishers Inc., 1990.

付 録

性能評価に使用したエフェクトの実例を付録としてここに紹介する。



図 A-1 うずまきエフェクト

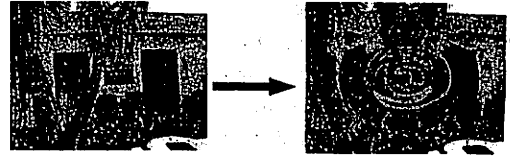


図 A-2 波紋エフェクト

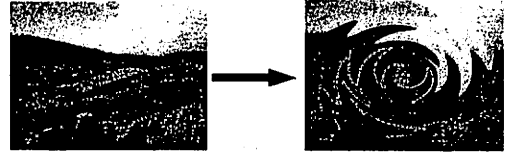


図 A-3 ジグザグエフェクト

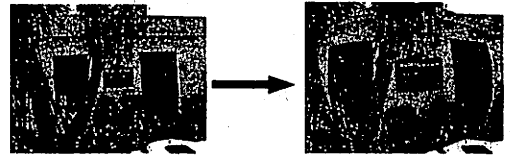


図 A-4 球面エフェクト

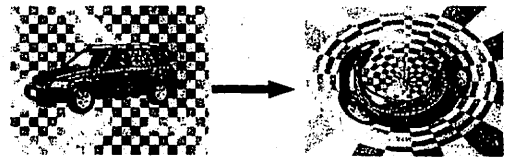


図 A-5 直交座標→極座標変換エフェクト

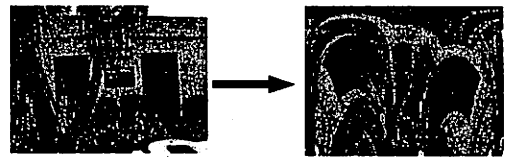


図 A-6 極座標→直交座標変換エフェクト