

構造可変型ビデオレート画像処理システム『韋駄天』

IDATEN: A Real-Time Image Processing System with Modifiable Network

佐々木 繁 佐藤 龍哉 岩瀬 洋道 後藤 敏行
Shigeru SASAKI Tatsuya SATO Hiromichi IWASE Toshiyuki GOTO

(株式会社 富士通研究所)
FUJITSU LABORATORY LTD.

Abstract - We developed a high-speed image processing system called IDATEN, which can continuously process time-varying images at video speed. The basic design concept was the improvement of the overall performance of the image processor system by adopting a new architecture. This paper proposes a "variable-structure pipeline" architecture, which uses a network to allow communication among any of the processing modules. We expanded the Benes multi-stage switching network to produce a flexible, high-speed pipeline processor. The experiments show the ease of programming and the effectiveness of this system.

key words - image processing, time-varying image, video rate, pipeline processor, variable structure, multi-stage switching network, switching algorithm.

1. まえがき

実際に動いている物体像を直接処理させることは、今日の画像処理システム研究者の夢の一つである。もちろん、ここでいう動く物体像の処理システムとは、画像の入力から出力までをITVカメラの走査速度つまり1秒間に30枚の速度(ビデオレート)で実行でき、しかもこの速度を低減させることなく、多種多様な画像処理に柔軟に対処できる仕掛けを持つことを意味する。

さて、この超高速性と柔軟性を実現するためのシステム・アーキテクチャとして、これまでに完全並列型、マルチプロセッサ型およびパイプライン型などが提案されているが、まだ多種多様な画像処理をビデオレートで実行できる段階には致っていない〔1〕～〔4〕。

この理由は様々挙げられるが、完全並列型とマルチプロセッサ型はメモリ上の画像データの処理を基本としているために、メモリ管理などに対するソフトウェアの存在する比率が極めて高く、入出力まで含めたシステム全

体のスループットを考えると、必ずしもプロセッサの持つ高速性が活かされない。一方、パイプライン型はプロセッサを直列に配置して処理を行うため柔軟性に欠けるが、画像をデータ列として処理するためにシーケンシャルなデータ転送を行う現行の入出力装置との整合性が良く、システム全体として的高速化が期待できる。

このような観点から、筆者らは高速化が見込めるパイプライン型アーキテクチャを基本とするが、その欠点である柔軟性の問題を補い、かつ高速性を活かした新アーキテクチャ「構造可変型パイプライン」を提案するとともに、このアーキテクチャに基づいたビデオレート画像処理システム『韋駄天』を開発した。

2. アーキテクチャ

2.1 問題点

パイプライン型アーキテクチャの特長は、ソフトウェアによる種々の制御管理を行うことなしに、画像処理を

実行させることができるため、高速化が期待できることにある。原理的には、システム・コンポーネントとしての各種画像処理プロセッサを直列に並べればよいので、完全並列型やマルチプロセッサ型に比べてビデオレートシステム実現の可能性が高い。

しかし、問題は柔軟性への対処である。図1は基本原理に基づく直接接続構成例であるが、この構成のもとで柔軟性を得るためには、各プロセッサが今日提案されている画像処理アルゴリズム全てをビデオレートで実行できるようにしなければならない、非現実的である。これに対処するために、最近ではプロセッサの機能を積和演算程度に限定し、それを図2のようなリング接続構成にすることで柔軟性を得ようとする試みがある〔5〕。しかし、この構成では、プロセッサ間の結合関係を変えようとするとデータ・バス上でデータ間の衝突に伴う待ちが生じ、高速処理が達成できないという課題を持つ。



図1. 直列接続構成 (高速、非柔軟)

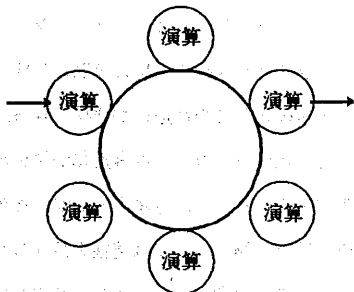


図2. リング接続構成 (中速、柔軟)

2.2 解決策

第2.1節で述べた現状の問題を解決するために、筆者らは、

- ・柔軟性：パイプライン結合の組み換え
- ・高速性：処理要素 (PM: Processing Module) のパイプライン結合

を柱としたアーキテクチャ「構造可変型パイプライン」を構築した。

図3はその基本モデルであり、柔軟性を実現するためのネットワーク機構と高速度な画像処理を実行するための複数のPMから成っている。このアーキテクチャは、ビデオレートでITVカメラやVTRから入力された画像データがネットワークと相互結合したPMを通過する間に、PMの処理機能に応じた画像処理を実行できるように工夫したものである。特にネットワークは、様々な画像処理に対処するために、PM同士の任意のパイプライン結合はもちろん、複数のパイプライン結合の形成ならびにフィード・バックを含んだパイプライン結合などが容易に実現できるようにした。

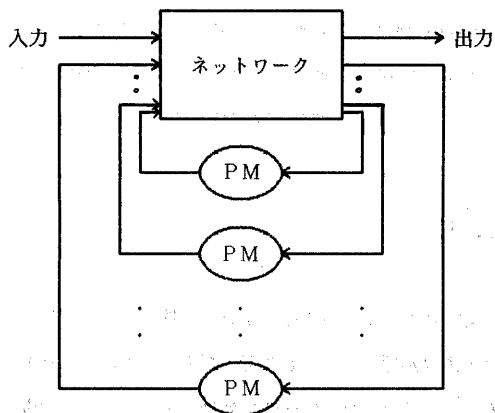


図3. 構造可変型パイプラインの原理

3. 「章駄天」

3.1 システム構成

図4は「章駄天」のシステム構成である。ホスト・プロセッサ部は、CPUとしてMC68000 (8MHz)、主記憶1.5MB、フロッピー・ディスク (1MB×2)、ハード・ディスク (160MB)、ユーザ端末 (最大10回線) から成っている。画像処理プロセッサ部は、ネットワークとそれに直結した複数のPM、IM (Image Memory) およびDM (Delay Module) から成っている。これらホスト・プロセッサ部と画像処理プロセッサ部とはマルチバスを介し

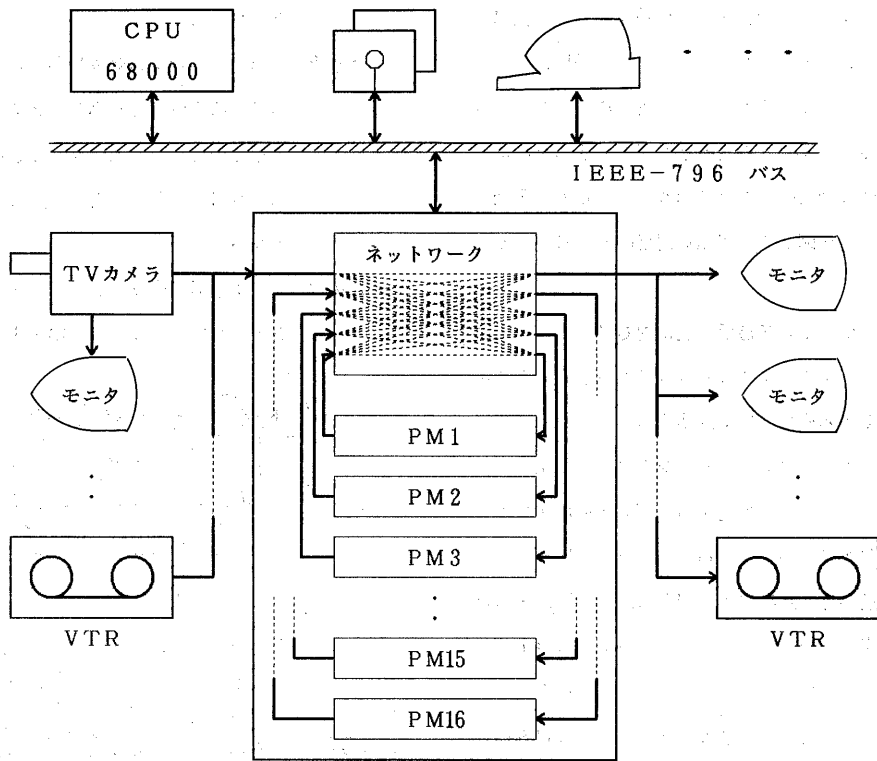


図4. 「章駄天」のシステム構成

て接続されており、また、各種ビデオ機器は画像処理プロセッサ部内のネットワークに直結されている。なお、画像処理プロセッサ部内のネットワーク制御用レジスタやPMのパラメータ・レジスタ、さらにはIMのアドレスなども、全てCPUの主記憶上に割り付けてられており、CPUから直接細かな制御を行うことができる。

3.2 柔軟性への対処・・・ネットワーク

ネットワークに関しては、これまで情報の高速転送や情報の適切な分配を可能とするようなスイッチング論理が数多く提案されている(6)。しかしながら、画像の重ね合わせや画像間の差分など複数枚の画像を対象とした画像処理特有の1対多通信伝送を簡単な論理で、かつコンパクトな回路規模で実現できる手段は今のところ存在しない。

そこで筆者らは、1対1通信ではあるが比較的小規模

な回路構成で実現できる多段スイッチング・ネットワーク Benes Permutation Network に着目し(7)、それを画像処理特有の1対多通信伝送が実現できるように改良した。改良点は二つある。

第一はネットワークを構成するスイッチング論理をこれまでの2状態から4状態にしたことである。図5はそのスイッチング論理であるが、直進、交換に加えて画像

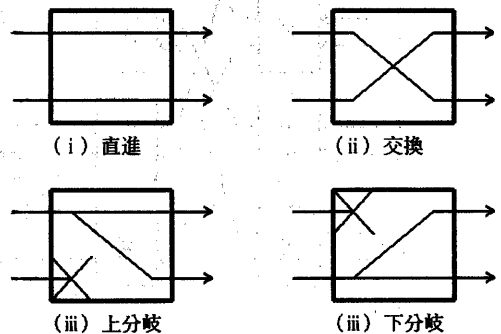


図5. スイッチ論理の状態

データの分配用として上分岐、下分岐を設けた。実際のネットワークは16×16のスイッチング論理によって構成したが、各スイッチング素子の実現においては、回路規模を小さく、また、高速転送を可能にするために、各素子を3個のPAL20X8だけで構成し、しかも素子間のパイプライン化を図ることで転送速度20MB/secを達成した。

第二の改良は、1対多通信伝送に対処できるネットワークのセッティング・アルゴリズムを考案したことである。以下にその内容について説明する。

図6をネットワークのセッティングによって求めたい遷移のフロー・グラフと仮定する。図6中の数値はネットワークに対する入力ノード番号であり、矢印はネットワークの接続関係を意味する。図7は図6を導出するための手順を表しているが、 i は入力ノード番号($i=1\sim 8$)、 $S(i)$ はセッティングによって並べ変えたい出力ノード番号である。つまり、図7のような組み換えを行えば、入力ノード列{0, 1, 2, 3, 4, 5, 6, 7}が、{5, 0, 7, 1, 0, 2, 4, 1}= $S(i)$ となり、図6のフロー・グラフが実現できる。実際には、つぎの5段階の処理によって実現した。

Step 1. 出力側に接続関係を持たないノード番号(ダミー・ノード)を見つける。図7では $i=3, 6$

が見つかる。

Step 2. $S(i)$ の中で複数回出現するノード番号に対してその1個だけを残し、他をダミー・ノードに置き換える。これによって、図7の $S(i)$ は、 $S(i) = \{5, 3, 7, 1, 0, 2, 4, 6\}$ となる。

Step 3. 1対1通信伝送セッティング・アルゴリズムをもとにしてスイッチング論理を決定する。

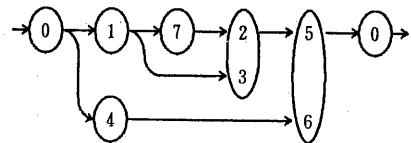


図6. フロー・グラフ

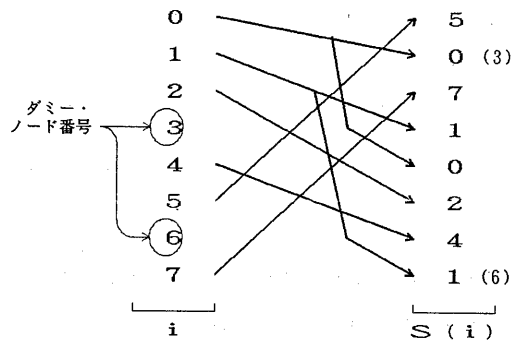


図7. ネットワークの生成

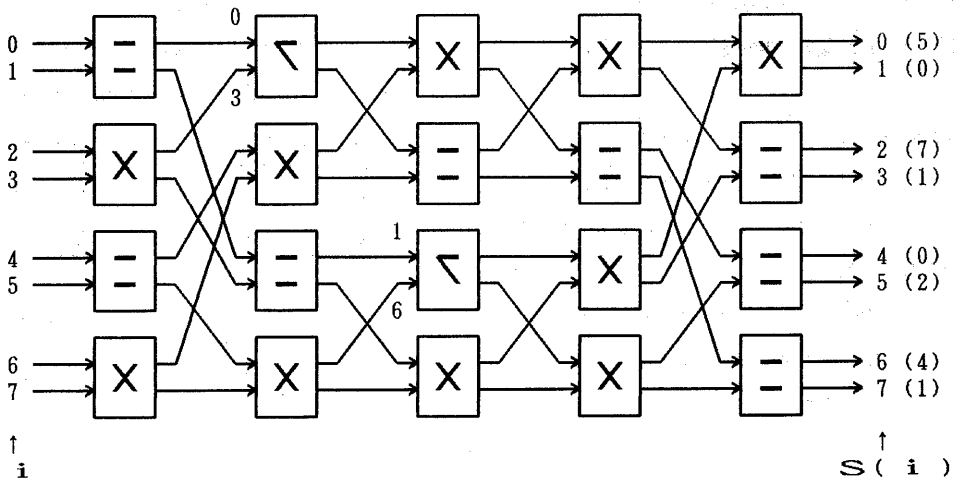


図8. 決定されたネットワークのスイッチング論理

これによりネットワークは、直進と交換の2状態で仮設定される。

Step 4. 各スイッチ毎に、Step 2 において複数個の中から残されたノード番号と、置き換えられたダミー・ノードとの対を検出する。検出されない場合には Step 2 へ戻り、ダミー・ノードの置き換え場所を変更することで同様の処理を繰り返す。

Step 5. 対が検出されたスイッチに対して、そのスイッチング論理を図5に示した分岐状態に再設定する。図7では0と3、1と6とが対として検出され、分岐状態になる。

図8は以上のアルゴリズムに基づいて、図6を実現するために決定されたネットワークのスイッチング論理である。「章駄天」ではノード数が16まで扱えるようになっており、しかもノード数を2の中乗単位で拡張できる。なお、ネットワークは画像データ(8ビット)とビデオ制御信号(4ビット)とを同時に伝送できる構成になっている。

3.3 高速性への対処・・・画像処理プロセッサ

3.3.1 PM (Processing Module)

これまで筆者らは各種画像処理を実行するプロセッサとして、局所並列による基本演算器を開発した〔8〕。「章駄天」の開発ではこの中から、論理フィルタ、空間フィルタ、画像間演算、濃度変換、ヒストグラム算出な

どの機能を取り上げ、それらをビデオ機器の同期信号で動作できるようにした。図9は、3×3のウィンドウで実現した空間フィルタ・モジュールのブロック図である。

3.3.2 IM (Image Memory)

IMは512×512(8ビット)の画像データをビデオレートで格納することのできるメモリである。このメモリは、各種画像間の演算や動画像における時間的な変化を検出する際のバッファとして利用できる。

3.3.3 DM (Delay Module)

DMもメモリであるが、これはPMがネットワークによって様々な構造でパイプライン化される際に生ずる画素単位の遅延を補正する機能を持つ。例えば、図10のように複数のパイプライン結合が形成された時、各パイプラインにおけるPMの個数やPM自身の持つ演算速度の相違により、パイプライン間に微妙な遅延ずれが生ずる。図10では遅延ずれを経路Aでd1、経路Bでd2となっており、DMを|d1-d2|分だけ遅延の少ない経路へ挿入することになる。「章駄天」では、予め全

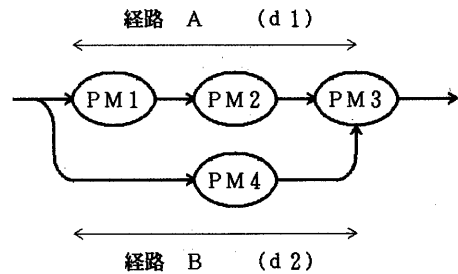


図10. 並行パイプラインにおけるDMの使用例

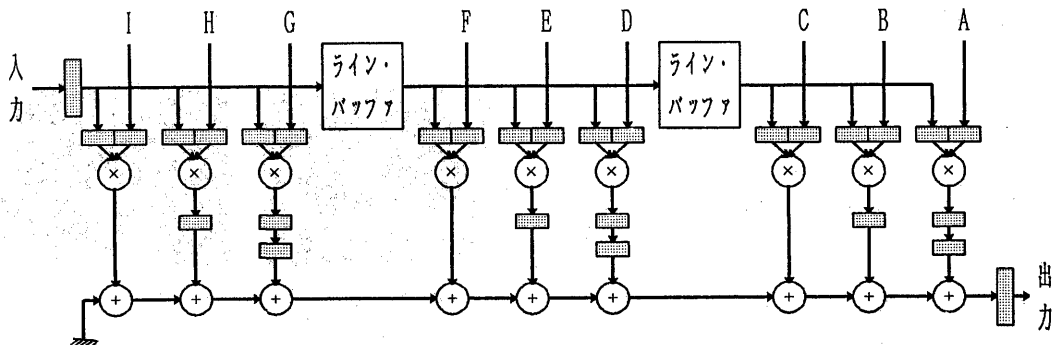


図9. 空間フィルタ・モジュール (A~I: 荷重係数)

でのPMの演算時間を記憶しておき、ネットワークが決定されれば自動的にDMでの遅延数が設定できるようになっている。実際のDMは画像データとビデオ制御信号から成る12ビットの情報を1単位とし、最大16384単位までの遅延が調整できる。

3.4 主要諸元

表1は「章駄天」の主な機能仕様であり、ネットワークの切り換えだけで16個の処理モジュールを自在にパイプライン結合でき、画像はそのパイプライン状に配列された処理モジュールによってビデオレートで処理される。また、ビデオ信号の垂直ブランキング期間に、動的にネットワークを切り換えることができるので、より高度なビデオ・レート処理も実行できる。

表1. 機能仕様

項目	諸元	
処理画像	512×512×8ビット	
処理速度	1画面当たり1/30秒	
パイプライン化	最大16個のプロセッサ	
ネットワーク	再帰完全型多段スイッチ 16×16 (12ビット幅)	
プロセッサ	種類	A/D、D/A、 空間フィルタ、論理フィルタ、濃度変換、 画像間演算、ヒストグラム算出、 フレーム・メモリ、ディレイ・モジュール
	演算速度	1画素当たり100nsec
接続可能なビデオ機器	TVカメラ、モニタ、VTR	

4. 実験結果

ここでは「章駄天」の高速性と有効性を検証するために、動いている物体をITVカメラから直接入力し、様々な画像処理を行った実例を紹介する。図11はある時刻($\tau=0$)における対象画像であり、この中で動く物体は「電車」と「人形」である。

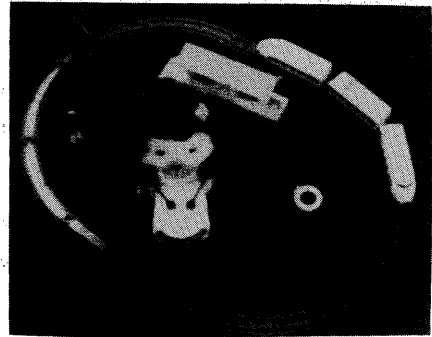


図11. 原画像 ($\tau=0$)

(1) 動画二値処理 (図12)

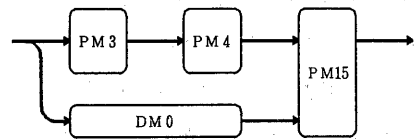
・ITVカメラから連続して入力されてくる8ビットの画像をある閾値で二値化し、更にその画像から輪郭成分を抽出した後、原画像と重ね合わせる。

・図12(a)はこの画像処理を実行するためのフロー・グラフであり、次のようなコマンドを入力すればよい。

```
net (0 3 4 15 0) (0 8 15)
```

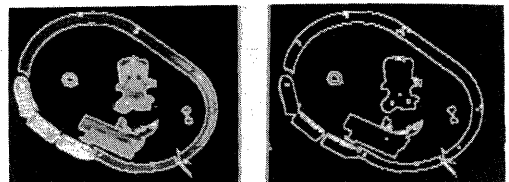
ここで各数値は各PMに対応する番号である。

・図12(b)は時刻 $\tau=0$ で二値化と輪郭抽出を行った結果および1/30秒後にそれと原画像を重ね合わせた結果である。



DM0 : ディレイ・モジュール
PM3 : 濃度変換
PM4 : 3×3 論理フィルタ
PM15 : 画像間演算

(a) フロー・グラフ



輪郭抽出 ($\tau=0$) 重ね合わせ ($\tau=1/30$)

(b) 処理結果

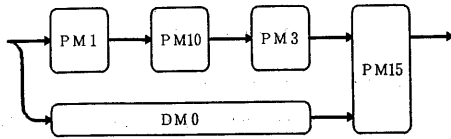
図12. 動画二値処理

(2) 動画濃淡処理 (図 1 3)

- ・ I T Vカメラから連続して入力されてくる8ビットの画像を符号付きデータに変換し、空間フィルタによりエッジ検出した後、原画像と重ね合わせる。
- ・ 図 1 3(a)はネットワーク構造であり、次のようなコマンドを入力すればよい。

net (0 1 10 3 15 0) (0 8 15)

- ・ 図 1 3(b) エッジ検出の結果と重ね合わせにより画像強調された結果である。

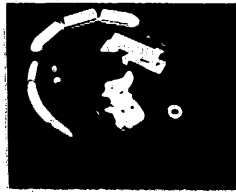


DM0 : デレイ・モジュール
 PM1,3 : データ変換 (符号変換)
 PM10 : 3×3 空間フィルタ
 PM15 : 画像間演算

(a) フロー・グラフ



エッジ (τ=45/30)



重ね合わせ (τ=45/30)

(b) 処理結果

図 1 3. 動画濃淡処理

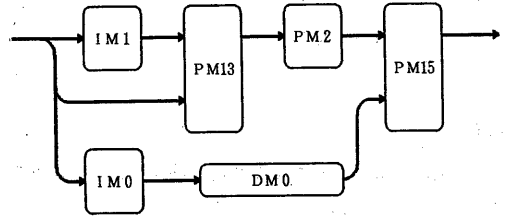
(3) 動画時間処理 (図 1 4)

- ・ I T Vカメラから連続して入力されてくる8ビットの画像から動きのある成分だけを抽出し、それを二値化するとともに、更にある時刻において数フレーム前の原画像と重ね合わせる。
- ・ 図 1 4(a)はネットワーク構造であるが、これを実現するためには3つのパイプラインに対応した次のコマンドを入力すればよい。

net (0 6 13 2 15 0) (0 13)

(0 5 8 15)

- ・ 図 1 4(b)は動き成分を $\tau=15/30$ で検出、二値化した結果と、および、 $\tau=92/30$ の結果と $\tau=20/30$ の原画像とを重ねた結果である。



DM0 : デレイ・モジュール
 IM0,1 : 画像メモリ
 PM2 : 濃度変換 (強調)
 PM13 : 画像間演算 (差分)
 PM15 : 画像間演算 (論理OR)

(a) フロー・グラフ



移動物体 (τ=15/30)



重ね合わせ (τ=92/30)

(b) 処理結果

図 1 4. 動画時間処理

5. むすび

本論文では、画像をビデオレートで処理できる新しいアーキテクチャを提案するとともに、それに基づいて試作したビデオレート画像処理システム「章駄天」について述べ、更に、実験によりその有効性を示した。

- ・ アーキテクチャでは、画像処理への柔軟性と高速性のいずれにも対処させるために、

”柔軟性 ⇒ パイプラインの組み換え”、

”高速性 ⇒ パイプライン化”、

を基本とした「構造可変型パイプライン」方式を考案した。

- ・ 「章駄天」では、プロセッサ間の自由なパイプライン化を可能とする再帰完全型多段スイッチング・ネットワークの実現並びに、局所並列による高速処理プロセ

ッサを実現することで、入力された画像を1画素当たり100nsecで自在に処理できるようになった。
実験では、動画を対象とした二値処理、濃淡処理及び時間的な処理などを通じて、画像がビデオレートで処理されることを実証した。

謝辞

日頃御指導いただく峯島部長、棚橋部長、吉田室長、並びに関連各位に深謝する。

参考文献

- (1) 木戸出, 坂上, "パイプライン方式と完全並列型が増えた最近の画像処理装置", 日経エレクトロニクス, pp.179-212, 7.19.1982.
- (2) Duff, M. J. B., "COMPUTING STRUCTURES FOR IMAGE PROCESSING", ACADEMIC PRESS (London), 1983.
- (3) Batcher, K. E., "Design of Massively Parallel Processor", IEEE Trans. on Comp., C-29, pp.836-840, 1980.
- (4) Gemmer, P., Ischen, H. and Luetjen, K., "FLIP: A Multiprocessor System for Image Processing", Languages and Architectures for Image Processing, pp.-245-256, ACADEMIC PRESS INC. (NewYork)LTD., 1981.
- (5) 天満, 溝口, 花木, "画像処理プロセッサTIPの構成とシミュレーション評価", 電子通信学会画像工学研究会報告, IE81-6, 1981.4.
- (6) Siegel, H. J., et al., "A Survey of Interconnection Methods for Reconfigurable Parallel Processing Systems", AFIPS, vol.48, pp.529-542, June 1979.
- (7) Nassimi, D. and Sahni, S., "Parallel Algorithm to Set Up the Benes Permutation Network", IEEE Trans. on Comp., Vol.C-31, No.2, 1982.
- (8) 山本, 榊井, 佐々木, 中鉢, "多機能な演算器と豊富なアルゴリズムライブラリを備えた画像処理システム", 電子通信学会論文誌D, Vol. J68, 1985.4.

「章駄天」命名の由来: 章駄天は護法神として知られており、世の中を指導統率するだけでなく非常に足の速い神様である。開発した画像処理システムが、多様な画像を臨機応変に処理でき、しかもそれをビデオレートで実行できることから「章駄天」と命名した。