

ARQUITETURA MACIÇAMENTE PARALELA PARA A ESTIMAÇÃO DE MOVIMENTO ATRAVÉS DA CORRELAÇÃO DE BLOCOS

Fernando Rosa do Nascimento
Instituto de Informática - UFRGS
Cx. Postal 15064, 91501-970 - Porto Alegre, RS
E-mail: fernando@inf.ufrgs.br

Abstract

This article present a high parallel architecture to solve the motion estimation problem by the exhaustive correlation between blocs technique. This problem is actually because the world intention to develop as fast as possibly a high definition television (HDTV). The developed architecture is a systolic type and it is proposed to solve in real time the problem, without penalizing in the memory organization to sustain the data flow to the processors.

Key-words

Block matching estimation, parallel architecture, systolic architecture, HDTV, line memory.

Resumo

Neste artigo é apresentado uma arquitetura maciçamente paralela para resolver o problema de estimação de movimento através da técnica de correlação exaustiva de blocos. Este problema é bastante atual visto o desejo mundial de se implantar o mais rapidamente possível uma televisão de alta definição (HDTV). A arquitetura desenvolvida é do tipo sistólica e se propõe a resolver em tempo real o problema proposto, sem com isto penalizar a organização de memória necessaria a manter o fluxo de E/S dos processadores.

Palavras chaves

Estimação pela correlação de blocos, arquiteturas paralelas, arquiteturas sistólicas, HDTV, memória de linha.

1. Introdução

A televisão como temos hoje foi concebida há mais de meio século e não atende mais à diversos aspectos técnicos como: recuperação de erros devido à poluição eletrônica crescente dos meios de radio-difusão, definição da imagem frente a CRTs de até 43 polegadas, etc. . Existe portanto uma necessidade de se implantar um novo sistema de televisão que atenda aos vários problemas e novos desafios. O principal é o de se usar os mesmos canais de hoje para se transmitir um novo padrão de televisão que tenha ao menos quatro vezes mais resolução que os atuais. Isto implica em uma banda de passagem do canal maior, além do que deve-se transmitir dados que foram digitalizados, resultando numa explosão da banda original.

As técnicas convencionais de compressão de dados (linerares) não são suficientes. A técnica aqui aplicada trabalha com uma compressão

de imagens considerando as relações temporais e espaciais que blocos de imagens "adjacentes" apresentam. É uma técnica adequada para imagens de televisão, onde de uma imagem para outra não existe muita alteração (na TV temos trinta imagens completas por segundo).

A estimação de movimento através da correlação exaustiva de blocos exige, na compressão, uma quantidade formidável de cálculos, que são basicamente as comparações de blocos, pixel à pixel, entre todos os blocos de uma imagem e os blocos da próxima imagem. Além disto, o bloco a ser testado deve ser comparado numa região de blocos e não simplesmente com um só bloco. Para uma imagem com tamanho de $576 * 720$ pixels e com 25 imagens por segundo são necessários 3,0 Goperações por segundo, só para o cálculo das operações de comparação.

Não existe hoje um processador seqüencial capaz de atender a esta demanda e o uso de processadores paralelos de alto desempenho apresentam um altíssimo custo. A solução é desenvolver uma arquitetura paralela especializada. Os processadores SIMDs são indicados para este trabalho, principalmente se funcionarem sob a forma sistólica. A arquitetura aqui apresentada foi desenvolvida para atender aos limites superiores dos casos da compressão em HDTV.

A segunda parte deste artigo apresenta uma noção aos algoritmos para estimação de movimentos através da correlação exaustiva de blocos. São apresentados as funções que definem o número de operações por segundo necessários ao algoritmo, a taxa de dados de entrada e a de saída de dados. A terceira parte apresenta dois exemplos onde todos os valores são analisados em função do tempo execução dos processadores, para máquinas de tamanhos diferentes.

A quarta parte descreve os princípios de funcionamento da arquitetura sistólica proposta para em seguida descrever a arquitetura da máquina e por fim de cada processador. A quinta parte apresenta a arquitetura de uma memória local que tem a finalidade de reduzir os valores da taxa de entrada de dados a matriz de processadores.

2. Noções sobre a correlação de blocos exaustivo

A imagem de uma televisão, a exemplo do cinema, é formada por uma sucessão de imagens fixas, chamadas de quadros, numa velocidade tal que o espectador tem a impressão de uma total continuidade das cenas (dos movimentos). No caso das televisões na América, a frequência de exibição (" f_r ") é de 30 quadros por segundo. Na televisão, durante a maior parte do tempo, existe uma grande correlação espacial entre as partes da imagem (blocos) de quadros que se sucedem. Esta característica deste tipo de sinal pode ser explorada para criar altas taxas de compressão, o principal requisito para se implantar sistemas de televisão de alta-definição (HDTV).

A estimação de movimentos é fundamentada na correlação entre blocos da imagem atual ("imagem de referência") e da imagem precedente ("imagem candidata"). Uma imagem de largura L e altura H é dividida em blocos de tamanho $n * n$ pixels, como visto na figura 1.

No caso das imagens de televisão, os blocos têm usualmente um tamanho de $8 * 8$ pixels ($n = 8$). O número de blocos de uma imagem é:

$$Q = (H * L) / n^2 \quad (1)$$

Cada bloco da imagem de referência deve ser pesquisado numa região da imagem candidata, definida no centro de um bloco de $n * n$ pixels, aumentado por deslocamento vertical e horizontal de $-p$ à $+p$ pixels (ver figura 1.b). Assim sendo o número de blocos candidatos é igual a:

$$q = (2p + 1)^2 \quad (2)$$

O bloco de referência ("BR") é um dos Q blocos (ver (1)) da imagem atual e os blocos candidatos ("BC") são os q blocos (ver (2)) da imagem precedente. O critério de comparação pode ser: o erro absoluto médio, o erro da média quadrática, a distância euclidiana e a intercorrelação normalizada. O critério mais simples para implementação em VLSI é o erro absoluto médio, o qual necessita fazer apenas uma subtração, o valor absoluto e a acumulação.

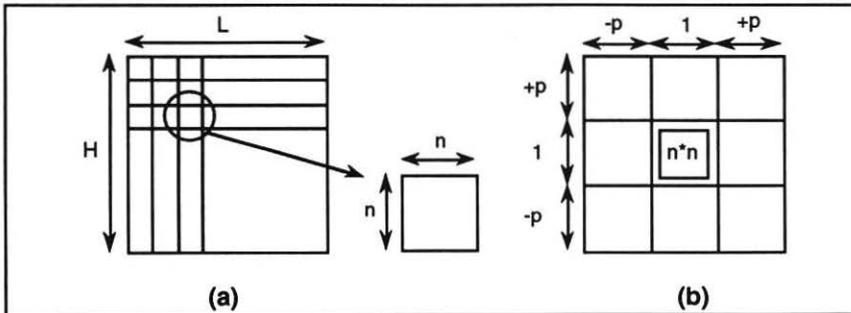


Figura 1 Uma imagem de dimensões $H * L$ pixels é dividida em blocos de $n * n$ pixels (a) e o espaço de pesquisa de um bloco de referência (b) contém $(2p + 1)^2$ blocos candidatos.

O somatório de $s(m, n)$ (ver (3)) das diferenças absolutas entre os pixels do BR (dado por $x(i, k)$) e os pixels do BC (dado por $y(i + m, k + n)$) é efetuada para cada BC (figura 1.(b)). O tratamento pelo critério do erro absoluto médio entre um BR e os BCs é calculado de forma seqüencial pela equação (3).

$$s(m, n) = \sum_{i=1}^N \sum_{k=1}^N |x(i, k) - y(i + m, k + n)|, \quad -p \geq m, n \geq p \quad (3)$$

$$\mu = \min_{m, n} \{s(m, n)\} \quad (4)$$

$$v = (m, n)_{\mu} \quad (5)$$

No cálculo de s a atualização dos índices m e n é feita a cada novo BC. Para cada BR existem q BCs, portanto temos q vetores v (dado por (5)) e o melhor deles é função de μ (ver (4)). O valor μ escolhido vai fornecer um vetor v associado, mas o valor de μ pode ter uma correlação

inaceitável entre o BC encontrado e o BR, isto é, o valor é maior do que um μ_{min} arbitrado para a aplicação. O vetor v fornece o "endereço" do BC encontrado por μ .

O algoritmo de correlação de blocos apresentado é classificado como exaustivo ("BMA exaustivo") pois para cada BR todos os BCs são pesquisados. Este algoritmo é caracterizado pelo elevado número de cálculos e pela velocidade do fluxo dos dados exigidos. São fatores importantes para estas características o valor da f_r e o tamanho da imagem ($H * L$ pixels). O número de vetores a serem calculados por cada imagem é dado por Q (ver (1)).

Para cada comparação de bloco (ver (3)) devem ser efetuados n^2 operações dos tipo $\sum |a - b|$. Para cada BR existem q BCs e para cada imagem tem-se Q BRs. O número total de cálculos necessários por unidade de tempo (" Op ") é definido por:

$$Op = n^2 \cdot Q \cdot q \cdot f_r = H \cdot L \cdot q \cdot f_r \quad (6)$$

Portanto o número total de cálculos é independente do tamanho do bloco ($n * n$). Op é diretamente proporcional a H, L, f_r e sobretudo ao deslocamento máximo (p) no espaço de pesquisa dos BCs. Chamaremos de $F_v = H * L * f_r$ como frequência de vídeo e a $T_v = 1 / F_v$ o período de entrada por pixel. A taxa de entrada (" R_{in} ") é o número de pixels necessários para tratar uma seqüência de imagens durante um segundo, calculada por:

$$R_{in} = (((2 \cdot p + n)^2 + n^2) \cdot Q \cdot f_r) = 10 \cdot H \cdot L \cdot f_r \quad (7)$$

A aproximação dada corresponde a $p=n$ (em geral $p = n = 8$). As equações (6) e (7) podem ser expressas em função de F_v :

$$Op = q \cdot F_v \quad \text{e} \quad R_{in} = 10 \cdot F_v \quad (8)$$

A taxa de saída do estimador de movimentos (" R_{out} ") é função do número de blocos por imagem (ver (1)) e do número de imagens por segundo (f_r). O vetor de deslocamento v tem um tamanho de $\log_2((2p+1)^2)$ bits, mas aqui supomos que seu tamanho é igual a uma palavra:

$$R_{out} = (H \cdot L / n^2) \cdot f_r \quad (9)$$

Pode-se observar que a relação entre R_{in} e F_v vale 10 (se $p = n$), portanto existe um grande potencial de reutilização de dados. No caso de uma máquina paralela os dados poderiam ser melhor aproveitados. Mas o valor de Op é fixo e independente da arquitetura.

3. Exemplos da necessidade computacional

A fim de exemplificar os requisitos de velocidade de cálculo e de dados numa aplicação de tratamento de vídeo, escolhemos como exemplo dois tamanhos de imagens, um referente ao formato de televisão atual e outro a HDTV. Nos dois casos f_r vale 30 Hz e $p = n = 8$. Os diversos valores são mostrados na figura 2, para os dois casos.

O requisito mais severo é a velocidade de cálculo por processador. Para a TV normal, $T_{256} = 93,76$ ns, tempo suficiente para um processador implementado com tecnologia moderna realizar a operação

$\sum |a-b|$ (num processador tradicional são necessários três instruções). No segundo caso (HDTV) se tivermos só dezesseis processadores, $T_{16} = 1,47$ ns, o que pode ser atendido com tecnologias de alta velocidade (ECL, AsGa, etc.).

H * L (pixels)	Q (bl/f _r)	F _v (Mp/s)	T _v (ns/pel)	Op (Gop/s)	R _{in} (Mp/s)	R _{out} (Kv/s)	T ₁₆ (ns)	T ₂₅₆ (ns)
525*600	4922	9,45	105,82	2,73	94,50	147,66	5,86	93,76
1050*1200	19688	37,80	26,45	10,92	378,00	590,63	1,47	23,44

Figura 2 Dois exemplos de tamanho de imagens para determinação dos valores de Op, R_{in}, T₁₆ e T₂₅₆.

Como os valores de R_{in} são elevados, precisa-se de uma organização da memória adequada capaz de atender as necessidades de dados dos processadores. Os algoritmos de estimação de movimentos através da correlação exaustiva de blocos pode necessitar uma arquitetura bastante cara, em função do tamanho da imagem e da f_r. Pode-se escolher outros exemplos em [KoPi89], mas a implementação pode ser muito mais complexa, sobretudo em termos de fluxo de dados.

Este algoritmo pode ser eventualmente calculado através de processadores especializados no tratamento de imagens (um apresentação dos mais significativos é feita em [Nas93]. No entanto nenhuma delas pode realizar com mais eficiência este algoritmo do que um processador paralelo projetado especialmente para esta tarefa.

4. Proposição de uma arquitetura VLSI dedicada

A arquitetura proposta tem por objetivo de atender aos requisitos do BMA exaustivo para a HDTV, em tempo real. De acordo com a tabela da figura 2, só uma máquina com 256 processadores teria condições de atender tais requisitos com o uso de tecnologia moderna e acessível. A solução proposta, em linhas gerais, é uma organização de bi-dimensional na qual cada processador elementar ("PE") se encarrega de fazer a comparação serial entre o BR e um determinado BC, resolvendo ainda o problema de fluxo de dados (R_{in}), o qual é mantido ao mínimo.

Esta nova arquitetura tem 256 PEs, onde segundo a forma de repartição proposta, cada um faz a comparação entre o BR e um dos 256 BCs. Portanto o tempo de cálculo de cada BR fica reduzido a n^2 pulsos de relógio (64 para $n = 8$). A figura 3 mostra uma parte da imagem onde cada bloco (BC) tem o tamanho de $8 * 8$ pixels (para $n = p = 8$). Para exemplificar o BR tem coordenadas (4,4) e o espaço de pesquisa da imagem anterior ((3,3), (3,4), (3,5), (4,3), (4,4), (4,5), (5,3), (5,4), (5,5)). O tratamento de cada coluna de um BR exige oito pulsos de relógio, onde um PE recebe um novo pixel (do BC) e envia ao próximo PE o pixel já tratado. O fluxo de dados dentro da matriz se dá em etapas, verticalmente para tratamento da coluna (oito ciclos) e horizontalmente para nova coluna (no início dos oito ciclos).

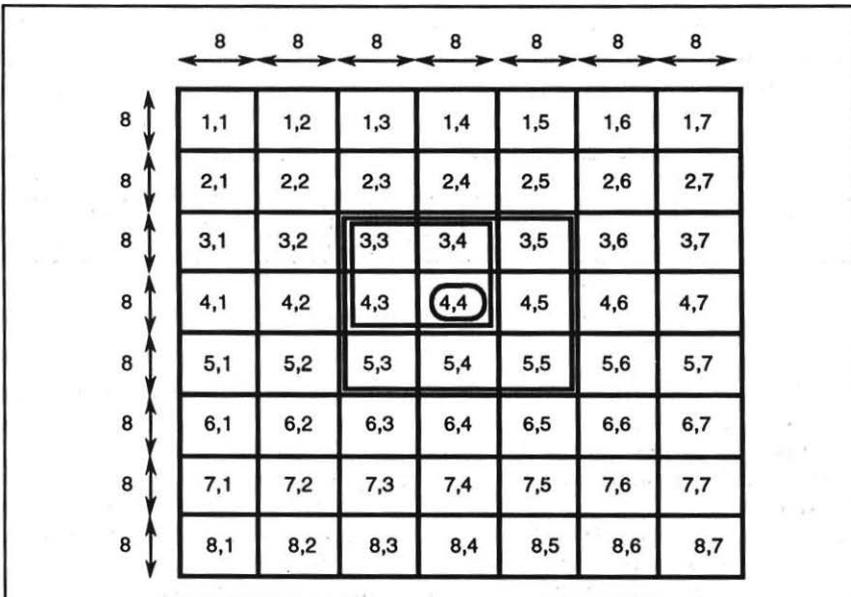


Figura 3 O mapeamento dos 256 processadores é feito no espaço de pesquisa, de tal forma que cada PE compare o seu BC com o BR atual.

No exemplo da figura 3 existem 256 BCs e cada um está associado a um PE. O valor de q é de 256 pois adotou-se p de -7 à +8 (ver equação (2)). O movimento de fluxo (e de cálculo) na horizontal é da esquerda para direita e na vertical de cima para baixo. Inicialmente a matriz de PEs vai estar "posicionada" nos blocos (3,3), (3,4), (4,3) e (4,4). Neste primeiro instante os PEs comparam todos os primeiros pixels dos BCs com o primeiro pixel do BR. No próximo ciclo todos os PEs se "deslocam" um pixel para baixo, melhor dizendo, os BCs são deslocados um pixel para cima. As comparações e o deslocamento continuam até se atingir o fim da coluna (de 8 pixels), neste instante os PEs se "deslocam" à direita, melhor dizendo, os BCs são deslocados para esquerda. Como os dados da coluna mais a esquerda dos PEs não serão mais necessários, eles são descartados.

Quando termina o cálculo do BR_{4,4}, a matriz de PEs vai "estar" exatamente sobre os blocos (3,4), (3,5), (4,4) e (4,5), que é o espaço de pesquisa para o novo BR_{4,5}. Assim o fluxo de dados se passa de forma contínua e os dados são reutilizados muitas vezes, reduzindo-se assim os requisitos de fluxo de dados. As arquiteturas bidimensionais são mais complexas em função justamente da necessidade de se assegurar um equilíbrio entre a potência de cálculo as taxas de entrada e saída de dados.

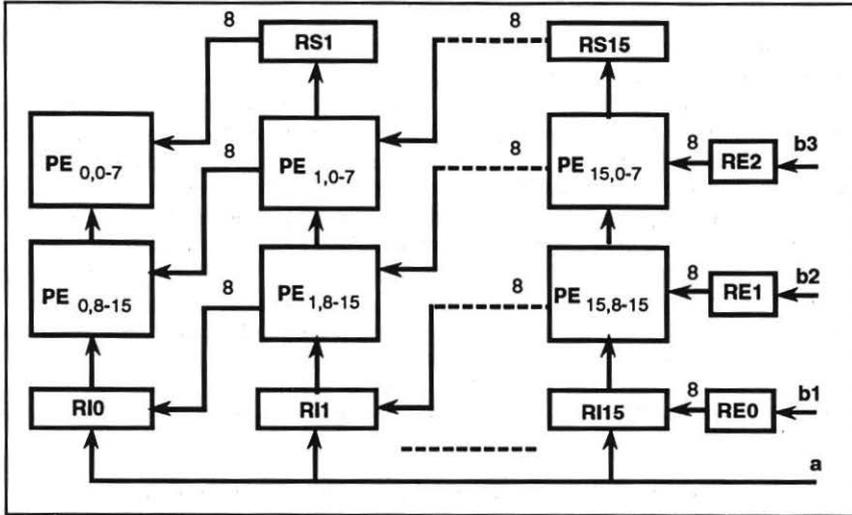


Figura 4 Arquitetura matricial com 256 PEs para efetuar o BMA exaustivo, incluindo os registradores para paralelização de dados (REs) e armazenamento temporário de dados (RIs).

A arquitetura proposta é vista na figura 4, e parte do princípio que cada PE deve comparar um BC com o BR, de forma seqüencial, isto é, como $n = 8$, demora 64 ciclos para efetuar o cálculo. No entanto temos 256 PEs que trabalham inteiramente em paralelo, assim ao fim de 64 ciclos todos os BCs foram calculados. O principal problema foi manter o fluxo de dados de forma coerente, mantendo-se as necessidades de dados ao mínimo. Para compreendermos seu funcionamento é necessário analisar a organização de cada PE.

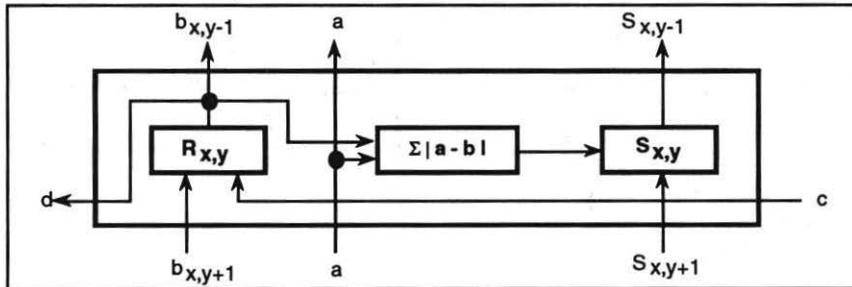


Figura 5 Organização interna de um processador elementar (PE) seus registros R, S e o acumulador das diferenças absolutas.;

A arquitetura de um *PE* é visto na figura 5. O processador é composto de uma unidade de cálculo dedicada (realiza a função $\sum |a-b|$) e de dois registradores. A unidade de cálculo faz a comparação pixel a pixel, recebe pela linha "a" o pixel do BR e da saída de $R_{x,y}$ o pixel do BC. O registrador $R_{x,y}$ forma um registro de deslocamento dos pixels dos BCs (deslocamento na vertical). Quando a coluna completou os 8 deslocamentos verticais, os dados são transferidos a coluna seguinte de *PEs* via a linha "d". Ao fim de 64 comparações tem-se o valor acumulado das diferenças de cada pixel em cada *PE*, e este valor é transferido ao seu respectivo registrador $S_{x,y}$.

Os registradores $S_{x,y}$ têm o trabalho de auxiliar no cálculo do melhor BC, deslocando estes dados para "fora" da matriz de processadores, de forma que outro circuito dedicado forneça o resultado final. Esta "rede" é essencialmente formada por comparadores que devem trabalhar numa primeira fase escolhendo em cada coluna o melhor BC e finalmente na horizontal o vetor do BC que tem mais correlação com o BR.

5 Memória local para reduzir a taxa de entrada de dados

Embora a matriz de processadores tenha sido concebida para reduzir ao mínimo o fluxo de dados necessários, ela precisa ainda de 4 pixels a cada ciclo de máquina (ver figura 4), a saber: "a" fornece o pixel do BR, "b1", "b2" e "b3" devem cada um fornecer um pixel que corresponde a uma banda de imagem (banda com largura de 8 pixels). Caso se adote somente esta arquitetura, podemos ver pela figura 3 que os dados dos BRs são solicitados três vezes. Por exemplo, no caso do bloco (5,4) o mesmo é solicitado para calcular os BRs (4,4), (5,4) e (6,4).

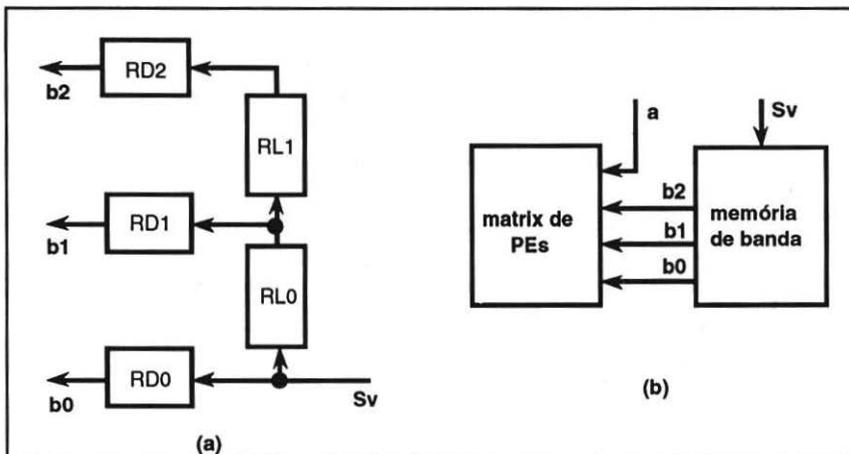


Figura 6 Memória local de duas bandas com largura de um bloco (a) e sua associação com a matriz de 256 *PEs* (b).;

A solução deste problema de demanda tríplice de dados pode ser resolvida com a implementação de uma memória local capaz de armazenar duas bandas de blocos. A implementação desta memória é semelhante a um registro de deslocamento, com comprimento de duas linhas e com largura de oito pixels (ver figura 6(a)).

Só existe necessidade de armazenar duas bandas de pixels pois o sinal "b0" é fornecido pelo próprio sinal S_V que alimenta toda a memória local. Como os dados de uma linha são deslocados simultaneamente com seu uso para os registros de linha (RLs), existe um reaproveitamento das bandas. Assim o conjunto da matriz de PEs com a memória local (ver figura 6(b)) só precisa de dois pixels a cada ciclo de máquina. Estes dois pixels são fornecidos a entrada "a" (para o BR) e a entrada S_V (para os BCs). Esta é a redução máxima que se pode obter partindo-se da idéia de que a imagem de referência e a imagem candidata estão numa memória principal.

6 Conclusões

Foi apresentado uma arquitetura maciçamente paralela capaz de processar o algoritmo de BMA exaustivo, em tempo real, para imagens do tipo HDTV. A implementação é custosa em número de PEs mas os problemas de compressão de imagens atualmente são maiores para as fontes geradoras de imagens (estações de TV), onde o benefício justifica o investimento. A regularidade da estrutura dos PEs facilita sua implementação, assim como a organização da memória local proposta.

Esta arquitetura foi simulada à partir dos modelos de sistemas digitais desenvolvidos em [Nasc93] na linguagem SIGNAL [Bour93]. Os resultados foram o esperado em termos funcionais, mas não foi realizado uma avaliação que envolvesse o desempenho e a qualidade final de imagem (de difícil avaliação).

A arquitetura projetada é dedicada e não permite alterações de parâmetros. A memória local pode ser programada, com acréscimo de uma pequena lógica de controle e roteamento.

Bibliografia

- [Bour93] Patricia Bournai et al. SIGNAL manual. Publication Interne 745, IRISA, Université de Rennes I, Rennes, France, julho 1993.
- [KoPi89] T. Komarek et P. Pirsch. Array architecture for block matching algorithms. *IEEE Transactions on Circuits and Systems*, 36(10):1301-1308, outubro 1989.
- [Nasc93] Fernando R. Nascimento. Méthodologie de conception d'architectures spécialisées - une étude de cas. Tese, Université de Rennes I, outubro 1992.