

ARQUITETURA DE HARDWARE PARA ESTIMAÇÃO E COMPENSAÇÃO DE MOVIMENTO UTILIZANDO PIXEL FRACIONÁRIO DO PADRÃO H.264/AVC

SANCHEZ, Gustavo Freitas; SAMPAIO, Felipe Martin; NOBLE, Diego Vrague; PORTO, Marcelo Schiavon; AGOSTINI, Luciano Volcan
GACI – Grupo de Arquiteturas e Circuitos Integrados
Universidade Federal de Pelotas

1 INTRODUÇÃO

Com o crescimento na demanda por vídeos de alta resolução, o esforço em pesquisa para desenvolvimento de algoritmos que possibilitassem maiores taxas de compressão também cresceu. Com isto, muitos padrões de compressão de vídeo foram desenvolvidos, sendo que o H.264/AVC é o mais novo e mais eficiente padrão de compressão de vídeo atualmente (RICHARDSON, 2003).

O principal objetivo do H.264/AVC é aliar altas taxas de compressão de vídeo a uma alta qualidade de vídeo. Para isto, é necessário o uso de algoritmos cada vez mais complexos. Esta complexidade impede, pelo menos na tecnologia atual, que estes algoritmos codifiquem o vídeo em tempo real (24 a 30 quadros por segundo) em software. Assim, soluções em hardware são necessárias para a codificação de vídeo de alta resolução em tempo real.

Este trabalho faz parte do esforço brasileiro para o desenvolvimento de soluções em *hardware* para o Sistema Brasileiro de TV Digital (SBTVD) (SBTVD, 2010), já que o H.264/AVC foi escolhido como o padrão do SBTVD.

Para codificar vídeos de forma eficiente, as informações redundantes, contidas em um vídeo, devem ser exploradas e reduzidas. Existem três tipos de redundâncias: temporal (similaridade entre quadros vizinhos), espacial (similaridade dentro do próprio quadro) e, entrópica (redundância na codificação binária).

A estimação de movimento (EM) é o módulo responsável por explorar a redundância temporal entre quadros vizinhos. Para fazer isto, o quadro que se deseja codificar é dividido em vários blocos, onde cada bloco é comparado com blocos de quadros já codificados. Quando o bloco mais parecido for encontrado, então é gerado um vetor de movimento, apontando para ele. Embora um vídeo seja formado por amostras de luminância e crominância, na EM, apenas amostras de luminância são utilizadas.

Para definir o quanto um bloco é similar a outro é utilizado um critério de similaridade. Neste trabalho será utilizado a Soma das Diferenças Absolutas (SAD) como critério de similaridade (RICHARDSON, 2003).

Um algoritmo de busca é utilizado para percorrer o quadro de referência em busca do melhor casamento de blocos. Entre os algoritmos mais conhecidos estão o *Full Search* (FS) (RICHARDSON, 2003) que testa todas as possibilidades em busca do melhor casamento e encontra o vetor ótimo, e o *Diamond Search* (DS) (PORTO, 2008) que é um algoritmo sub-ótimo que encontra bons casamentos, porém não gera resultados ótimos.

O DS segue os padrões apresentados na Figura 1, onde L representa um ponto do *Large Diamond Search Pattern* (LDSP) e S representa um ponto do *Small Diamond Search Pattern* (SDSP) (PORTO, 2008).

Inicialmente, são comparados os pontos do LDSP, aplicando o SAD entre os blocos destes pontos e o bloco do quadro atual. Caso a maior similaridade não

seja encontrada no centro, o algoritmo é reaplicado utilizando a posição que obteve a maior similaridade como o novo centro do LDSP. Quando a maior similaridade for encontrada no centro, então é feito um refinamento, onde os pontos do SDSP são comparados e então é gerado um vetor de movimento para a melhor posição.

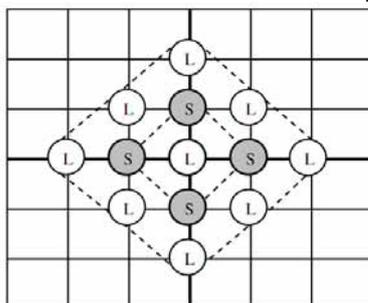


Figura 1 – Padrões do *Diamond Search*

Devido a suas características este algoritmo atinge uma qualidade inferior ao FS, porém o DS consegue realizar a busca em uma área maior e em um intervalo de tempo menor, reduzindo drasticamente o número de operações de cálculos de SAD.

A compensação de movimento (CM) é o módulo responsável por remontar o quadro codificado. Normalmente a CM é realizada após a estimação de movimento. A CM busca, na memória principal do codificador, os blocos apontados pelos vetores de movimento gerado na EM e, então, monta o quadro reconstruído, e então calcula a diferença do quadro reconstruído em relação ao quadro original sem codificação. Esta diferença será somada ao quadro reconstruído no processo de decodificação. Neste trabalho, a diferença calculada quando o critério de similaridade é aplicado é armazenada em bancos de registradores. Desta forma, não é necessário realizar a CM para as amostras de luminância, agilizando a codificação do vídeo e reduzindo o número de acessos a memória principal do codificador.

Apesar da EM ser responsável por grande parte dos ganhos em compressão dos codificadores atuais, seus resultados podem ser melhorados com a utilização de pixels fracionários. Através da interpolação de pixels ao redor do vetor escolhido é possível gerar novas posições intermediárias, e então, verificar estas posições criadas, gerando um vetor fracionário. Estes vetores podem ser com precisão de quarto ou de meio pixel (*quarter/half pixels*). Para facilitar a integração com uma EM fracionária, este trabalho tem como saídas, além do vetor de movimento, também as informações da área ao redor do vetor escolhido que servirá como entrada para a EM fracionária, diminuindo os acessos a memória principal do codificador quando a busca fracionária for realizada.

Este trabalho apresenta uma arquitetura para a estimação de movimento para blocos 8x8 utilizando o algoritmo *Diamond Search* e, gera as amostras para a EM fracionária e realiza a compensação de movimento integrada para as amostras de luminância. O principal objetivo é reduzir o número de acessos a memória principal do codificador, já que estas etapas geralmente são realizadas após a estimação de movimento, exigindo novos acessos à memória principal. Com a solução apresentada neste artigo, os dados são buscados da memória uma única vez e, após serem usados na EM, são reutilizados para a CM e para a EM fracionária.

2 METODOLOGIA

A arquitetura desenvolvida neste trabalho foi descrita em VHDL (PEDRONI, 2004) e sintetizada em um FPGA da Altera (ALTERA, 2010). A Figura 2 apresenta a arquitetura desenvolvida. Inicialmente, as memórias estão vazias, então, a memória atual e a memória de referência são preenchidas com os dados lidos da memória externa, que contém o vídeo original sem compressão. Após isto, os dados da memória de referência são colocados em memórias locais, cada uma contendo os dados que serão utilizados em cada unidade de processamento (UP), que é responsável pelo cálculo da similaridade entre os blocos (SAD).

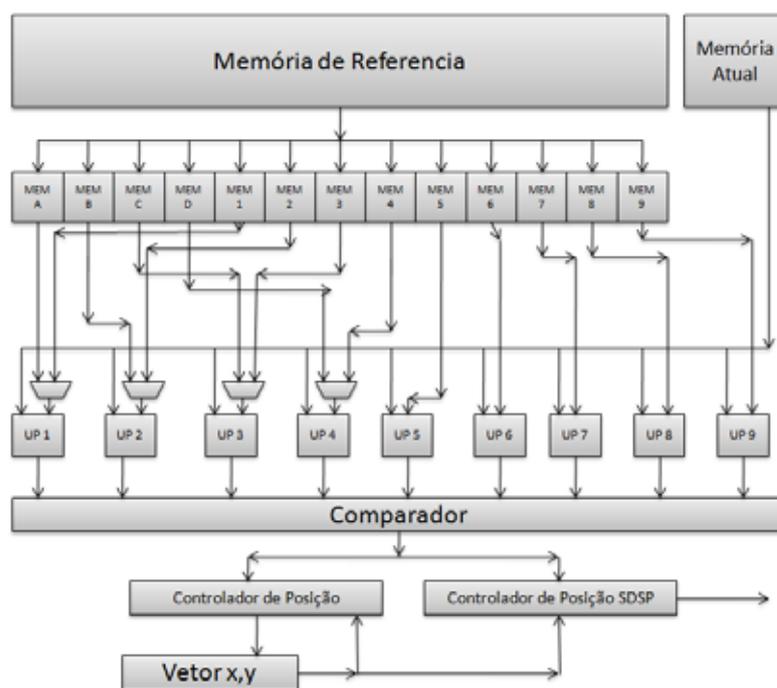


Figura 2 – Arquitetura Desenvolvida

As memórias numeradas de 1 até 9 (MEM 1 a MEM 9) contém os dados para o cálculo do SAD para o LDSP. As memórias de A até D (MEM A a MEM D) contém os dados para o refinamento SDSP. Após as memórias locais estarem cheias, os dados são enviados para as UPs onde é calculado o SAD.

O comparador é responsável por encontrar qual bloco obteve o menor SAD (maior similaridade), e então, o vetor é atualizado para uma nova posição. Isto é repetido até que o melhor resultado seja encontrado no centro (na UP 5) e, então, o controlador de posição SDSP gera o vetor de movimento final.

Dentro das 5 primeiras UPs ainda existem bancos de registradores que são responsáveis por armazenar os resíduos, assim não sendo necessário realizar a CM para as amostras de luminância.

3 RESULTADOS E DISCUSSÕES

A arquitetura foi sintetizada para o dispositivo EP4S100G2F40I1 da família Stratix 4 da Altera (ALTERA, 2010). Os resultados da síntese são apresentados na Tabela 1.

Tabela 1 – Resultados de Síntese da arquitetura desenvolvida

Frequencia Máxima	ALUTs	Registradores	Bits de Memória
256,28 MHz	7310	9045	8384

A taxa de processamento da arquitetura desenvolvida, considerando um número máximo de cinco iterações do LDSP, e a frequência de operação obtida na síntese, é apresentada na Tabela 2.

Tabela 2 – Taxa de processamento da arquitetura desenvolvida

Resolução	Quadros por Segundo
FULL HD (HDTV 1080p)	49,4
HDTV 720p	111
EDTV 480p	250

Para a codificação de vídeo em tempo real é necessário que pelo menos de 24 a 30 quadros sejam codificados por segundo. A solução arquitetural desenvolvida pode operar em tempo real para vídeos de alta resolução, como HDTV 1080p (Full HD).

4 CONCLUSÕES

Este trabalho apresentou o desenvolvimento de uma arquitetura para a EM com a CM integrada que gera as amostras para posteriormente serem interpoladas e ser possível realizar uma busca de até quarto de pixel.

A arquitetura proposta atingiu uma frequência máxima de 256,28 MHz, e é capaz de processar até 49,4 quadros Full HD por segundo. A solução arquitetural desenvolvida apresentou um baixo consumo de recursos de hardware, cerca de 6% das células lógicas disponíveis no dispositivo.

A arquitetura desenvolvida pode processar um número maior de quadros do que o necessário para a codificação em tempo real, desta forma, é possível diminuir sua frequência de operação, contribuindo para a redução do consumo de energia durante o processo de codificação.

5 REFERÊNCIAS

ALTERA CORPORATION: Altera - The Programmable Solutions Company. Disponível em: <<http://www.altera.com>> Acesso em: Agosto 2010.

RICHARDSON, I. **H.264/AVC and MPEG-4 Video Compression – Video Coding for Next-Generation Multimedia**. Chichester: John Wiley and Sons, 2003.

FÓRUM DO SISTEMA BRASILEIRO DE TV DIGITAL TERRESTRE. Disponível em: <<http://www.forumsbtvd.org.br>> Acesso em: Agosto 2010.

PORTO, Marcelo. **Arquiteturas de Alto Desempenho e Baixo Custo em Hardware para a Estimação de Movimento em Vídeos Digitais, 2008**. Mestrado em Ciência da Computação. Instituto de Informatica – UFRGS, Porto Alegre, 2008.

PEDRONI, Volnei. **Circuit Design with VHDL**. Massachusetts: MIT Press, 2004.