

República Federativa do Brasil
Ministério do Desenvolvimento, Indústria
e do Comércio Exterior
Instituto Nacional da Propriedade Industrial.

(21) **PI0616034-4 A2**

(22) Data de Depósito: 13/09/2006
(43) Data da Publicação: 07/06/2011
(RPI 2109)



(51) *Int.Cl.:*
H04N 7/24 2011.01

(54) Título: **DECODIFICADOR DE SUBAMOSTRAGEM DE MUDANÇA DE RESOLUÇÃO ARBITRÁRIA**

(30) Prioridade Unionista: 14/09/2005 US 11/226.043

(73) Titular(es): MICROSOFT CORPORATION

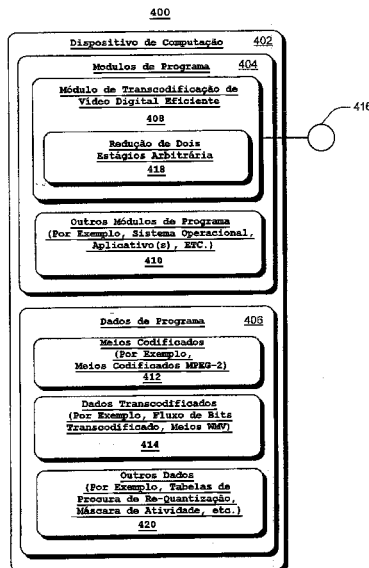
(72) Inventor(es): GUOBIN SHEN, SHIPENG LI, WANYONG CAO, YUWEN HE

(74) Procurador(es): NELLIE ANNE DANIEL SHORES

(86) Pedido Internacional: PCT US2006035939 de 13/09/2006

(87) Publicação Internacional: WO 2007/033346 de 22/03/2007

(57) Resumo: DECODIFICADOR DE SUBAMOSTRAGEM DE MUDANÇA DE RESOLUÇÃO ARBITRÁRIA A decodificação de subamostragem de mudança de resolução arbitrária é descrita. Em um aspecto, um fluxo de bits codificado é recebido. O fluxo de bits codificado é reduzido em um laço de decodificação no domínio DCT para gerar dados reduzidos.



"DECODIFICADOR DE SUBAMOSTRAGEM DE MUDANÇA DE RESOLUÇÃO ARBITRÁRIA"

Fundamentos da Invenção

Conteúdo de vídeo digital é tipicamente gerado para 5
ra mirar um formato específico de dados. Um formato de dados de vídeo geralmente obedece a um padrão de codificação de vídeo específico ou um algoritmo de codificação proprietário, com uma taxa de bits específica, resolução espacial, taxa em quadros, etc. Tais padrões de codificação incluem 10
MPEG-2 e WINDOWS Media Vídeo (WMV). A maioria do conteúdo de vídeo digital existente é codificada de acordo com o formato de dados MPEG-2. WMV é amplamente aceito como um codec qualificado no âmbito da transmissão contínua, sendo amplamente desenvolvido por toda a Internet, adotado pelo consórcio HD-
15 DVD, e atualmente sendo considerado como o padrão SMPTE. Diferentes padrões de codificação de vídeo fornecem capacidades de compressão e qualidade visual variáveis.

Transcodificação refere-se ao processo geral de converter um fluxo de bits comprimido em um outro fluxo comprimido. Para uma capacidade do dispositivo e redes de distribuição, ela é freqüentemente desejável para converter um 20
fluxo de bits em um formato de codificação em um outro formato de codificação como de MPEG-2 a WMV, para H.264, ou até para um formato escalável. A transcodificação pode também
25 ser utilizada para alcançar alguma funcionalidade específica tal como funcionalidade tipo VCR, inserção de logo, ou capacidade de elasticidade de erro aperfeiçoada do fluxo de bits para transmissão pelos canais sem fio.

A Fig. 1 mostra um sistema de Transcodificador no Domínio de Pixel em Cascata (CPDT), que coloca em cascata um decodificador auxiliar para decodificar um fluxo de bits de entrada com um codificador que gera um novo fluxo de bits com um conjunto de parâmetros de codificação diferente ou em novo formato. Uma desvantagem dessa arquitetura de transcodificação convencional é que sua complexidade tipicamente apresenta um obstáculo para o desenvolvimento prático. Como um resultado, a arquitetura de transcodificação CPDT da Fig. 1 é tipicamente usada como uma avaliação do desempenho para esquemas aperfeiçoados.

A Fig. 2 mostra uma arquitetura de transcodificador no domínio DCT em cascata (CDDT), simplificando a arquitetura CPDT da Fig. 1. O sistema da Fig. 2 limita a funcionalidade para mudanças de parâmetros de codificação e de subamostragem de resolução espacial/temporal. O CDDT elimina processos DCT/IDCT implementados pelo transcodificador CPDT da Fig. 1. Ainda, o CDDT executa MC no domínio DCT, que é tipicamente uma operação computacionalmente dispendiosa e que consome tempo. Isso ocorre porque os blocos DCT são frequentemente sobrepostos com blocos MC. Como um resultado, a arquitetura CDDT tipicamente necessita aplicar operações de matriz de ponto flutuante computacionalmente dispendiosas e complexas de modo a executar MC no domínio DCT. Adicionalmente, o refinamento de vetor de movimento (MV) é tipicamente impossível utilizando a arquitetura CDDT.

Sumário da Invenção

Este sumário é fornecido para introduzir uma sele-

ção de conceitos de uma forma simplificada que são adicionalmente descritos abaixo na descrição detalhada. Este sumário não pretende identificar características chave ou essenciais do assunto reivindicado, nem pretende ser usado como
5 uma ajuda na determinação do escopo do assunto reivindicado.

Em vista do acima, a decodificação de subamostragem de mudança de resolução arbitrária é descrita. Em um aspecto, um fluxo de bits codificado é recebido. O fluxo de bits codificado é reduzido em um laço de decodificação no
10 domínio DCT para gerar dados reduzidos.

Breve Descrição dos Desenhos

Nas figuras, o dígito mais à esquerda de um número de referência de componente identifica a figura particular na qual o componente primeiro aparece.

15 A Fig. 1 mostra um sistema de Transcodificador no Domínio de Pixel em Cascata (CPDT), que coloca em cascata um decodificador auxiliar para decodificar um fluxo de bits de entrada com um codificador para gerar um novo fluxo de bits com um conjunto de parâmetros de codificação diferente ou em
20 novo formato.

A Fig. 2 mostra uma arquitetura de transcodificador no domínio DCT em cascata (CDDT) convencional, simplificando a arquitetura CPDT da Fig. 1.

25 A Fig. 3 mostra uma arquitetura dividida de transcodificação no domínio de pixel não integrada exemplificada para transcodificar MPEG-2 para WMV, de acordo com uma modalidade. Mais particularmente, essa arquitetura dividida fornece uma base conceitual para transcodificação de vídeo di-

gital integrado eficiente.

A Fig. 4 mostra um sistema exemplificado para transcodificação de vídeo digital integrado eficiente, de acordo com uma modalidade.

5 A Fig. 5 mostra um transcodificador no domínio de pixel em cascata com laço fechado simplificado exemplificado, de acordo com uma modalidade.

10 A Fig. 6 mostra um transcodificador no domínio DCT com laço fechado simplificado exemplificado, de acordo com uma modalidade.

15 A Fig. 7 mostra uma operação de combinação exemplificada de quatro blocos DCT 4x4 em um bloco DCT 8x8, de acordo com uma modalidade. Essa operação de combinação é executada durante transcodificação de conteúdo de vídeo eficiente.

 A Fig. 8 mostra uma arquitetura exemplificada para um transcodificador de subamostragem de resolução 2:1 numérica no domínio DCT simplificado, de acordo com uma modalidade.

20 A Fig. 9 mostra uma operação de combinação exemplificada de quatro blocos DCT 4x4 em um bloco DCT 8x8 para meios de interlace para operações de transcodificação de subamostragem de resolução espacial 2:1, de acordo com uma modalidade.

25 A Fig. 10 mostra uma arquitetura de transcodificador de subamostragem de mudança de resolução arbitrária 2:1 simplificada exemplificada com completa compensação de desvio, de acordo com uma modalidade.

A Fig. 11 mostra um modelo de verificador de armazenador temporário virtual (VBV) padrão exemplificado para um decodificador.

A Fig. 12 mostra um transcodificador com redução de resolução espacial arbitrária, de acordo com uma modalidade.

A Fig. 13 mostra um procedimento exemplificado para operações de transcodificação de vídeo digital integradas eficientes, de acordo com uma modalidade.

A Fig. 14 mostra um ambiente exemplificado onde transcodificação de vídeo digital integrado eficiente pode ser parcial ou completamente implementada, de acordo com uma modalidade.

Para propósitos de discussão e ilustração, a cor é usada nas figuras para apresentar as seguintes convenções. Uma seta azul sólida representa sinal no domínio de pixel com relação aos dados de imagem real ou residual. Uma seta sólida vermelha representa o sinal no domínio DCT. Uma seta tracejada laranja representa informação de movimento.

Descrição Detalhada da Invenção

Visão Geral

Sistemas e métodos para eficiente transcodificação de vídeo digital são descritos abaixo em relação às Figs. 4 até 14. Esses sistemas e métodos utilizam informação no fluxo de bits de entrada para permitir que um aplicativo dinamicamente controle propagação de erro, e desse modo, seletivamente controle velocidade e qualidade de transcodificação de fluxo de bits de vídeo. Esse controle seletivo permite

que um aplicativo escale de forma inconsútil de esquema de transcodificação de laço fechado (perfil de transcodificação de alta velocidade) para esquemas de transcodificação de laço aberto (perfil de transcodificação de alta qualidade). Em
5 contraste a arquiteturas de transcodificação convencionais (por exemplo, o CPDT da Fig. 1 e o CDDT da Fig. 2), as arquiteturas para eficiente transcodificação de vídeo digital são integradas e combinam diferentes tipos de Transformadas Discretas de Cosseno (DCTs) ou transformadas tipo DCT em um
10 módulo de transcodificação. Os sistemas e métodos para eficiente transcodificação de vídeo implementam re-quantização com uma tabela de procura rápida, e fornecem bons mecanismos de controle de desvio usando um algoritmo de limite triplo.

Em uma implementação, onde eficiente transcodifi-
15 cação de vídeo digital transcodifica um formato de dados de fluxo de bits (por exemplo, MPEG-2, etc.) para WMV, as operações de transcodificação de perfil de alta qualidade suportam características de codificação avançadas de WMV. Em uma implementação, operações de transcodificação de perfil
20 de alta velocidade implementam redução em dois estágios de resolução arbitrária (por exemplo, quando transcodificando de alta definição (HD) para definição padrão (SD)) - por exemplo, tal como em um decodificador de subamostragem de mudança de resolução arbitrária. Em tais operações de redução
25 em dois estágios, parte da relação de redução é eficientemente alcançada no domínio DCT, enquanto as operações de relação de redução são implementadas no domínio espacial em uma resolução substancialmente reduzidas.

Base Conceitual Exemplificada

A Fig. 3 mostra a arquitetura dividida de transcodificação no domínio de pixel em cascata não integrada exemplificada 300 para converter MPEG-2 em WMV. Essa arquitetura dividida não é integrada porque módulos separados respectivamente executam operações de decodificação e codificação. A arquitetura dividida da Fig. 3 fornece uma base conceitual para subsequente descrição dos sistemas e métodos integrados para eficiente transcodificação de vídeo digital. A TABELA 1 mostra símbolos e seus respectivos significados para discussão da Fig. 3.

TABELA 1

e_{i+1}	Erro de quadro (i+1) a ser codificado pela parte de codificação do transcodificador.
$\hat{B}_i$	Quadro reconstruído i pelo decodificador MPEG-2 em resolução original.
$\tilde{B}_i$	Quadro reconstruído i pelo codificador na resolução original.
$\hat{b}_i$	Quadro reconstruído i pelo decodificador MPEG-2 em resolução reduzida.
$\tilde{b}_i$	Quadro reconstruído i pelo codificador em resolução reduzida.
$\hat{r}_{i+1}$	Resíduos reconstruídos de quadro (i+1) pelo decodificador MPEG-2.
$\tilde{r}_{i+1}$	Resíduos reconstruídos de quadro (i+1) pelo codificador.

$MC_{mp2}(B, mv)$	Predição de movimento compensado com a imagem de referência B e o vetor de movimento mv pelo decodificador MPEG-2, com base em bloco 16x16.
$MC_{vc1}(B, mv)$	Predição de movimento compensado com a imagem de referência B e o vetor de movimento mv pelo transcodificador 308 (codificador), com base em bloco 16x16 ou 8x8.
$MC'_{mp2}(b, mv)$	Predição de movimento compensado com referência de resolução reduzida b e o vetor de movimento mv, usando filtragem MPEG-2, com base em bloco 8x8 ou menor.
$MC'_{vc1}(b, mv)$	Predição de movimento compensado com referência de resolução reduzida b e o vetor de movimento mv, usando filtragem de transcodificador 308, com base em bloco 8x8 ou menor.
MV	Vetor de movimento na resolução de quadro original.
mv	Vetor de movimento na resolução de quadro reduzida.

Para propósitos de descrição e ilustração exemplificada, o sistema 300 é descrito com relação à transcodificação de MPEG-2 para WMV com redução de taxa de bits, redu-

ção de resolução espacial, e sua combinação. Muitos conteúdos de vídeo digital existentes são codificados de acordo com o formato de dados MPEG-2. WMV é amplamente aceito como um codec qualificado no âmbito de transmissão contínua, sendo amplamente desenvolvido por toda a Internet, adotado pelo Consórcio HD-DVD, e atualmente sendo considerado como um padrão SMPTE.

MPEG-2 e WMV fornecem capacidades de compressão e qualidade visual variáveis. Por exemplo, as técnicas de compressão respectivamente usadas por MPEG-2 e WMV são diferentes. Por exemplo, as técnicas de filtragem por compensação de movimento (MC) e por precisão de vetor de movimento (MV) são diferentes. Em MPEG-2, a precisão de movimento é até precisão de meio pixel e o método de interpolação é filtragem bilinear. Em contraste, em WMV, a precisão de movimento pode ir até precisão de um quarto de pixel, e dois métodos de interpolação chamados de filtragem bilinear e filtragem bicúbica são suportados. O uso de WMV pode resultar em até uma redução de 50% na taxa de bits de vídeo com perda de qualidade visual desprezível, se comparada a uma taxa de bits MPEG-2.

Em um outro exemplo, as transformadas usadas por MPEG-2 e WMV são diferentes. Por exemplo, MPEG-2 usa padrão DCT/IDCT e o tamanho da transformada é fixado em 8x8. Em contraste, WMV usa transformadas inteiras (VCL-T) onde elementos da matriz kernel de transformada são todos inteiros pequenos. Adicionalmente, o tamanho da transformada pode ser alterado usando WMV de blocos em blocos usando ou 8x8, 8x4,

4x8 e 4x4. MPEG-2 não suporta otimização em nível de quadro. Onde, WMV suporta várias sintaxes em nível de quadro para otimização de desempenho. WMV suporta muitas outras características de codificação avançadas tal como compensação de
 5 intensidade, redução de faixa, e mudança de resolução dinâmica, etc.

Em vista do acima, para fornecer redução de taxa de bits sem mudança de resolução, o processo de filtragem ligando o decodificador MPEG-2 e o codificador WMV mostrados
 10 na Fig. 3 é um filtro passa-tudo (isto é, sem efeito). Portanto, a entrada para o codificador para o quadro (i+1) é expressa como:

$$e_{i+1} = \hat{r}_{i+1} + MC_{mp2}(\hat{B}_i, MV_{mp2}) - MC_{vc1}(\tilde{B}_i, MV_{vc1})$$

(1)

15 Nessa implementação, a eficiência de codificação WMV da Fig. 3 ganha resultado da precisão de movimento mais fina. Em WMV, a precisão de movimento de quarto de pixel é permitida junto à precisão de meio pixel comum como em MPEG-2. Além disso, WMV permite interpolação melhor, mas mais
 20 complexa conhecida como interpolação cúbica para filtragem MC. A interpolação bilinear é usada para MPEG-2 no módulo MC (MC_{mp2}) para MC de meio pixel. O método de interpolação bilinear similar àquele usado em WMV com a exceção de que a interpolação bilinear MPEG-2 não tem controle de arredondamen-
 25 to. Para alcançar alta velocidade, a precisão de meio pixel pode ser implementada na parte do codificador. Uma razão para isso é a ausência do quadro original absoluto (isto é, dados de entrada de fluxo de bits (BS_IN) já estão comprimi-

dos). Assim, nesse exemplo, é difícil obter um vetor de movimento mais preciso e ainda significativo. Por outro lado, a informação de movimento obtida a partir do decodificador MPEG-2 (isto é, $MV_{vc1} = MV_{mp2}$) pode ser reutilizada diretamente. Como não há mudança de resolução, não há perda de precisão MV com essa hipótese. Se o codificador é adicionalmente restrito ao uso de interpolação bilinear e força o parâmetro de controle de arredondamento para estar sempre desligado, então sob a hipótese razoável de que a compensação de movimento é uma operação linear e ignorando o erro de arredondamento (isto é, $MC_{vc9} = MC_{mp2}$), a Equação 1 é simplificada como segue:

$$e_{i+1} = \hat{r}_{i+1} + MC_{mp2}(\hat{B}_i - \tilde{B}_i, MV_{mp2})$$

(2)

De acordo com a Equação 2, o transcodificador CPDT de referência na Fig. 3 pode ser simplificado. Tal arquitetura simplificada é descrita abaixo em relação à Fig. 5. Antes de descrever a arquitetura simplificada, um sistema exemplificado para eficiente transcodificação de vídeo digital é primeiro descrito.

Um Sistema Exemplificado

Embora não exigida, a transcodificação de vídeo digital eficiente é descrita no contexto geral de instruções de programa de computador sendo executadas por um dispositivo de computação tal como um computador pessoal. Geralmente, módulos de programa incluem rotinas, programas, objetos, componentes, estruturas de dados, etc., que executam tarefas particulares ou implementam tipos de dados abstratos parti-

culares. Enquanto os sistemas e métodos são descritos no contexto seguinte, ações e operações descritas aqui podem também ser implementadas em hardware.

A Fig. 4 mostra um sistema exemplificado 400 para
5 eficiente transcodificação de vídeo digital. Nessa implementação, as operações do sistema 400 são descritas com relação a e esquemas de codificação de vídeo por compensação de movimento (MC) baseado em bloco e de DCT híbrido, mediante os quais muitos padrões de codificação de vídeo e formatos pro-
10 prietários são baseados. Mais particularmente, o sistema 400 é descrito com arquiteturas, componentes, e operações usados para transcodificar MPEG-2 em WMV. Entretanto, pode-se apreciar que as arquiteturas, componentes e operações descritos para transcodificação de eficiência e complexidade escalável
15 incorporada pelo sistema 400 para transcodificar MPEG-2 em WMV podem também ser aplicados a outras conversões de formato de dados de fluxo de bits além de MPEG-2 e WMV. Por exemplo, em uma implementação, o sistema 400 é utilizado para transcodificar o fluxo de bits MPEG-2 em fluxo de bits MPEG-
20 4 e os dados de fluxo de bits MPEG-4 em dados de fluxo de bits WMV, etc. Em tais modalidades alternativas, as seguintes arquiteturas de transcodificação descritas do sistema 400 (incluindo componentes e operações associadas a eles), consideram o tipo de dados de fluxo de bits sendo decodifi-
25 cados, codificados, e respectivos formatos de dados.

Nessa implementação, o sistema 400 inclui um dispositivo de computação de propósito geral 402. Este representa qualquer tipo de dispositivo de computação tal como um

computador pessoal, um laptop, um servidor, dispositivo de
 computação portátil ou móvel, etc. O dispositivo de computa-
 ção 402 inclui módulos de programa 404 e dados de programa
 406 para transcodificar um fluxo de bits codificado em um
 5 primeiro formato de dados (por exemplo, MPEG-2) em um fluxo
 de bits codificado em um formato de dados diferente (por e-
 xemplo, WMV). Os módulos de programa 404 incluem, por exem-
 plo, o módulo de transcodificação de vídeo digital eficiente
 408 ("módulo de transcodificação 408") e outros módulos de
 10 programa 410. O módulo de transcodificação 408 transcodifica
 meios codificados 412 (por exemplo, meios MPEG-2) em meios
 transcodificados 414 (por exemplo, meios WMV). Outros módu-
 los de programa 410 incluem, por exemplo, um sistema opera-
 cional e um aplicativo utilizando as capacidades de transco-
 15 dificação de fluxo de bits de vídeo do módulo de transcodi-
 ficação 408, etc. Em uma implementação, o módulo de transco-
 dificação 408 expõe suas capacidades de transcodificação ao
 aplicativo via uma Interface de Programação de Aplicação
 (API) 416.

20 Transcodificação de Perfil de Alta Velocidade

A Fig. 5 mostra um transcodificador no domínio de
 pixel em cascata de laço fechado integrado simplificado e-
 xemplificado sem propagação de erro. Para propósitos de dis-
 cussão e ilustração, os componentes da Fig. 5 são descritos
 25 em referência aos componentes da Fig. 4. Por exemplo, a ar-
 quitetura da Fig. 5 é representativa de uma implementação de
 arquitetura exemplificada do módulo de transcodificação 408
 da Fig. 4. Com relação à arquitetura 500 da Fig. 5, compara-

do à arquitetura na Fig. 3, nota-se que nessa está uma arquitetura integrada sem componentes de codificador e decodificar separados. Adicionalmente, nota-se que o módulo de estimativa de movimento de refinamento de MV é removido do MC no decodificador MPEG-2. Adicionalmente, MC no codificador WMV é combinado a um MC que opera em erros de re-quantização acumulados. Dessa maneira, a arquitetura de transcodificação da Fig. 5 reduz de forma significativa a complexidade de computação para transcodificação de alta velocidade de formatos de dados de vídeo progressivos e interlaçados.

Nota-se que a transformada WMV é diferente da usada em MPEG-2. Em MPEG-2, o ponto de flutuação padrão DCT/IDCT é usado se a transformada de inteiros, cuja propriedade de empacotamento de energia é parecida com DCT, é adotado em WMV. Como um resultado, o IDCT no decodificador MPEG-2 e o VC1-T no codificador WMV não cancelam um ao outro. A transformada de inteiros em WMV é diferente da implementação de inteiros de DCT/IDCT. A transformada de inteiros em WMV é cuidadosamente projetada com todos os coeficientes de transformada sendo pequenos inteiros. Transcodificadores convencionais não são integrados para transcodificar um fluxo de bits codificado com relação a uma primeira transformada em uma segunda transformada que não é a mesma da primeira.

A Equação 3 fornece uma matriz de transformada exemplificada para VC1-T 8x8.

$$T_8 = \begin{bmatrix} 12 & 12 & 12 & 12 & 12 & 12 & 12 & 12 \\ 16 & 15 & 9 & 4 & -4 & -9 & -15 & -16 \\ 16 & 6 & -6 & -16 & -16 & -6 & 6 & 16 \\ 15 & -4 & -16 & -9 & 9 & 16 & 4 & -15 \\ 12 & -12 & -12 & 12 & 12 & -12 & -12 & 12 \\ 9 & -12 & 4 & 15 & -15 & -4 & 16 & -9 \\ 6 & -16 & 16 & -6 & -6 & 16 & -16 & 6 \\ 4 & -9 & 15 & -16 & 16 & -15 & 9 & -4 \end{bmatrix}$$

(3)

A Equação 3 em combinação com as equações 4 e 5, que são descritas abaixo, indica como duas transformadas diferentes são implementadas em um componente de escala do módulo de transcodificação 408 (Fig. 4). Em uma implementação, a precisão de VC1-T é precisão de 16 bits, que é muito adequada para implementação MMX. Como um resultado, a complexidade do codec pode ser significativamente reduzida.

A Fig. 6 mostra um transcodificador no domínio DCT de laço fechado simplificado exemplificado. A arquitetura da Fig. 6 é representativa de uma implementação de arquitetura exemplificada do módulo de transcodificação 408 (Fig. 4). A arquitetura 600 da Fig. 6 é uma arquitetura simplificada se comparada à arquitetura 500 da Fig. 5. Com relação à Fig. 6, faz-se C_8 ser a matriz de transformada DCT padrão, B , o bloco DCT MPEG-2 quantizado inverso, e b , o IDCT de B , então o IDCT MPEG-2 é calculado como segue:

$$b = C'_8 B C_8$$

Faz-se $\hat{B}$ ser o VC1-T de b , então $\hat{B}$ é calculado como:

$$\hat{B} = T_8 b T'_8 \circ N_{88}$$

onde $\circ$ denota a multiplicação no sentido do ele-

mento de duas matrizes, e N_{88} é a matriz de normalização para transformada VC1-T que é calculada como segue:

$$N_{88} = c_8 \cdot c'_8$$

com

$$c_8 = \begin{bmatrix} 8/288 & 8/289 & 8/292 & 8/298 & 8/288 & 8/289 & 8/292 \\ 8/298 \end{bmatrix};$$

$\hat{B}$ é diretamente computado a partir de B, usando a seguinte fórmula:

$$\hat{B} = T_8(C'_8 B C_8) T'_8 \circ N_{88}$$

10 (4)

Para verificar que $T_8 C'_8$ e $C_8 T'_8$ estão muito próximos a matrizes diagonais, se aplica-se a aproximação, então a Equação 4 se torna uma escala no sentido do elemento da matriz B. Ou seja:

$$15 \quad \hat{B} = B \circ S_{88} \quad (5)$$

onde

$$S_{88} = \text{diag}(T_8 C'_8) \cdot \text{diag}(C_8 T'_8) \circ N_{88}$$

A Equação 5 mostra que o VC1-T no codificador WMV e o IDCT no decodificador MPEG-2 podem ser combinados. Con-
20 seqüentemente, a arquitetura na Fig. 5 pode ser adicionalmente simplificada à mostrada na Fig. 6. A comparação detalhada revela que os dois módulos DCT/IDCT são substituídos por dois módulos VC1-T e VC1-T inverso. Em uma implementação, um simples módulo de dimensionamento é também adiciona-
25 do. Dois comutadores são embutidos junto com uma máscara de atividade nessa arquitetura. Esses componentes embutidos, como descrito abaixo, são usados para controle dinâmico da complexidade de operações de revestimento de transcodifica-

ção do transcodificação 408 (Fig. 4). Nesse ponto, esses componentes são conectados. A propriedade aritmética de 16 bits da transformada WMV a leva a processamento paralelo para PC e DSP. Em vista disso, as complexidades de computação são significativamente reduzidas. Além disso, como todos os elementos da matriz de dimensionamento, S88, estão substancialmente próximos em relação um ao outro, essa computação, e uma implementação, é substituída por uma multiplicação escalar.

10 As Figs. 5 e 6 mostram respectivas arquiteturas de transcodificação de laço fechado exemplificadas, onde um laço de retorno é envolvido. Nessa implementação, o laço de retorno, que incluiu desquantização VC-1, transformada inversa VC-1, acúmulo de erro de resíduo e MC em erro acumulado, compensa o erro causado pelo processo de re-quantização VC-1. O erro de re-quantização é uma causa principal do erro de desvio para transcodificadores de redução de taxa de bits, tal como aquele mostrado na Fig. 1. Embora as arquiteturas de transcodificação das Figs. 5 e 6 não sejam completamente livres de desvio, mesmo com compensação de erro, o erro de desvio é muito pequeno. Isso ocorre porque a causa restante de erro de desvio é o erro de arredondamento durante filtragem por compensação de movimento. Um mérito de compensação de erro residual é que as arquiteturas das Figs. 5 e 6 fornecem o ligamento e o desligamento dinâmico do processo de compensação, como descrito abaixo com relação à TABELA 2. A arquitetura de transcodificação da Fig. 6 executada pura transcodificação de redução de taxa de bits de MPEG-

2 para WMV tal como conversão SD para SD ou HD para HD de uma maneira substancialmente ótima.

Mais particularmente, as arquiteturas de transcodificador em cascata convencionais (por exemplo, as arquiteturas das Figs. 1 e 2) carecem de flexibilidade de complexidade. Com relação às economias de computação, o máximo que tal arquitetura convencional pode alcançar é através de reutilização de MV e mapeamento de modo. Por outro lado, as arquiteturas de compensação de erro residual acumulada, por exemplo, a arquitetura da Fig. 6 (e as arquiteturas das Figs. 8 e 10, como descrito abaixo) tem escalabilidade embutida em termos de complexidade. A TABELA 2 mostra significados exemplificados de comutadores na Fig. 6.

TABELA 2

15 Comutadores Exemplificados para Controle Dinâmico de Velocidade e Qualidade de Transcodificação

S_0	Nível de bloco	Comutador de acúmulo de erro
S_1	Nível de bloco	Comutador de atualização de erro
S_2	Nível de bloco	Comutador de decisão de pular antes o bloco

Depois do módulo de transcodificação 408 da Fig. 4 ter implementado simplificação livre de desvio, um aplicativo pode dinamicamente compensar entre a complexidade e a

qualidade para acelerar a velocidade de transcodificação. Nessa implementação, a qualidade pode ser compensada para velocidade, e vice versa. Em outras palavras, algum erro de desvio pode ser permitido no transcodificador simplificado adicional. Com essa estratégia, o erro de desvio introduzido no método mais rápido é limitado e completamente controlável. Baseadas nessa consideração, três comutadores (S_0 , S_1 e S_2) são fornecidos nas arquiteturas das Figs. 6, 8 e 10. Os comutadores são usados somente para as arquiteturas baseadas em compensação de erro residual. Os comutadores seletivamente pulam algumas operações de consumo de tempo para reduzir a complexidade substancialmente, enquanto introduzindo somente uma pequena quantidade de erro. Os significados de vários comutadores são resumidos na TABELA 2. Decisões computacionais associadas com esses comutadores são eficientemente obtidas de acordo com critérios descritos abaixo com relação a cada comutador.

O comutador S_0 controla quando o erro de re-quantização de um bloco deveria ser acumulado no armazenador temporário de erro residual. Se comparado a um seletor de reconstrução padrão, a função do comutador S_0 é aperfeiçoada adotando-se um processo de re-quantização baseado em uma tabela de procura rápida e fornecendo-se um mecanismo de controle de desvio mais fino via um algoritmo de limite triplo. Como um resultado, todas as observações feitas com relação ao comutador S_0 são consideradas. Por exemplo, em uma implementação, a diferença de energia no domínio DCT pode ser utilizada como o indicador.

O comutador S_1 controla quando o módulo que consome mais tempo, MC do erro residual acumulado. Em uma implementação, o comutador S_1 está ligado. Uma máscara de atividade binária é criada para o quadro de referência. Cada elemento da máscara de atividade corresponde à atividade de um bloco 8×8 , como determinado por

$$Atividade(Bloco_i) = \begin{cases} 1, & Energia(bloco_i) > Th \\ 0, & Energia(bloco_i) \leq Th \end{cases}$$

onde $Energia(bloco_i)$ é a energia do bloco no armazenador temporário de erro residual acumulado. Em uma implementação, $Energia(bloco_i)$ é calculada no domínio espacial e no domínio DCT. $Energia(bloco_i)$ pode ser aproximada pela soma de valores absolutos. Se MV aponta para blocos pertencentes à área de baixa atividade, então MC do erro residual acumulado para esse bloco específico é pulado.

O comutador S_2 executa detecção anterior para determinar se o erro de bloco deveria ser codificado. Isso é especialmente útil em aplicativos de conversão de taxa de fluxo onde o codificador aplica um tamanho de etapa de quantização maior. Nessa implementação, se o sinal de entrada (a soma do MC do erro residual acumulado e o resíduo reconstruído a partir do decodificador MPEG-2) é mais fraco do que um limite, então o comutador S_2 é desligado tal que nenhum erro será codificado.

Em uma implementação, limites para os comutadores S_0 , S_1 e S_2 são ajustados tal que quadros de referência anteriores são processados com maior qualidade em menor velocidade. Isso ocorre porque o propósito dos comutadores é al-

cançar uma melhor compensação entre qualidade e velocidade, e por causa da natureza preditiva de codificação.

Transcodificador de Perfil de Alta Qualidade

Se a mudança de taxa de bits não é significativa ou a qualidade da fonte de entrada não é muito alta, a arquitetura da Fig. 6 substancialmente otimiza a redução da taxa de bits quando convertendo os fluxos de bits MPEG-2 em fluxos de bits WMV. Por outro lado, a fonte de entrada pode ser de alta qualidade e saída de alta qualidade pode ser desejada, também a velocidade de transcodificação pode ser uma exigência moderada (por exemplo, em tempo real). Um transcodificador de perfil de alta qualidade, tal como o transcodificador no domínio de pixel em cascata (CDPT) da Fig. 3 com refinamento MV, alcança esses critérios. Com essa arquitetura, pode-se ligar todas as características de codificação avançadas do codificador WMV para assegurar que eficiência de codificação mais alta pode ser alcançada.

Mudança de Resolução

Em sistemas de transcodificação de meios convencionais, há geralmente três fontes de erros para transcodificação com redução de resolução espacial. Esses erros são como segue:

- Redução: Erros gerados quando obtendo um vídeo reduzido. É tipicamente uma escolha enraizada quando projetando operações do filtro de redução para fazer uma compensação entre qualidade visual e complexidade, especialmente quando reduzindo no domínio espacial.

- Erro de Re-quantização: Como com o processo de

transcodificação de redução de taxa de bits pura, esse é o erro devido à re-quantização com um tamanho de etapa de re-quantização maior.

- Erro de MV: MV incorreto levará a predição compensada por movimento errada. Como um resultado, não importa como o erro de re-quantização é compensado, e não importa quão alta a taxa de bits é, um resultado perfeito é difícil de obter se não re-computando a compensação de movimento baseada nos novos MVs e modos. Esse é um problema para sistemas convencionais que transcodificam Quadros B, porque WMV suporta somente um modo MV para Quadros B. Esse poderia também ser um problema se um deseja executar otimização, que levaria a mudança de modo de codificação, por exemplo, de modo quatro MV para um MV. Além disso, o problema geralmente existe para componentes de crominância já que eles são tipicamente compensados com um único MV. (Esse não é um problema para as arquiteturas de transcodificação de vídeo digital eficiente descritas quando aplicada a Quadros P. Uma razão para isso é porque WMV suporta modo de codificação de quatro MV para Quadros P).

As operações do módulo de transcodificação 408 (Fig. 4) abordam as duas últimas fontes de erros, como agora descrito.

Compensação de Erro de Re-quantização

Faz-se D denotar a filtragem de subamostragem. Com relação à arquitetura da Fig. 3, a entrada para o codificador VC-1 para o quadro $(i+1)$ é derivada como segue:

$$e_{i+1} = D(\hat{r}_{i+1}) + D(MC_{mp2}(\hat{B}_i, MV_{mp2})) - MC_{vc1}(\tilde{b}_i, mv_{vc1}) \quad (6)$$

Assume-se que $MC_{vc1} = MC_{mp2}$, $mv_{mp2} = mv_{vc1} = MV_{mp2}/2$.

Com a aproximação que

$$D(MC_{mp2}(\hat{B}_i, MV_{mp2})) = MC'_{mp2}(D(\hat{B}_i), D(MV_{mp2})) = MC'_{mp2}(\hat{b}_i, mv_{mp2})$$

(7),

5 A Equação 6 é simplificada para o seguinte:

$$e_{i+1} = D(\hat{r}_{i+1}) + MC'_{mp2}(\hat{b}_i - \tilde{b}_i, mv_{mp2}) \quad (8)$$

O primeiro termo na Equação 8, $D(\hat{r}_{i+1})$, refere-se ao processo de redução do sinal de resíduo MPEG-2 decodificado. Esse primeiro termo pode ser determinado usando filtragem e decimação passa-baixa no domínio espacial. Entretanto, o uso de redução no domínio DCT para obter esse termo resulta em uma redução de complexidade e melhor PSNR e qualidade visual. Os resultados da redução no domínio DCT são substancialmente melhores do que resultados obtidos através da filtra-

10 gem bilinear no domínio espacial ou filtragem de 7 vezes no domínio espacial com coeficientes $(-1, 0, 9, 16, 9, 0, -1)/32$. Nessa implementação, a redução no domínio DCT retém somente os coeficientes DCT de baixa frequência 4x4 na esquerda superior. Ou seja, aplicar um IDCT 4x4 padrão nos co-

15 eficientes DCT retidos resultará em uma imagem espacialmente reduzida 2:1 (isto é, meios transcodificados 414 da Fig. 4).

O segundo termo na Equação 8, $MC'_{mp2}(\hat{b}_i - \tilde{b}_i, mv_{mp2})$, implica em compensação de erro de re-quantização em uma resolução reduzida. Nessa implementação, o MC no decodificador

20 MPEG-2 e o MC no codificador WMV são combinados em um único processo MC que opera em erros de re-quantização acumulados na resolução reduzida.

A Fig. 7 mostra uma operação de combinação exem-

plificada de quatro (4) blocos DCT 4x4 em um bloco DCT 8x8. Um problema prático permanece. Na redução no domínio DCT, quatro blocos DCT 8x8 (blocos, B_1 até B_4 , em um macrobloco MPEG-2 (MB) na resolução original) são mapeados para os quatro sub-blocos 4x4 de um bloco 8x8 do novo MB na resolução reduzida e ainda no domínio DCT (por exemplo, ver Fig. 7). Em WMV, Quadros P e Quadros B, o tipo de transformada 4x4 é permitido. Como um resultado, nada necessita ser feito adicionalmente exceto o dimensionamento acima mencionado. Entretanto, para Quadros I, somente o tipo de transformada 8x8 é permitido. Assim, quando lidando com Quadros I, o módulo de transcodificação 408 (Fig. 4) converte os quatro sub-blocos DCT de baixa frequência 4x4 em um bloco DCT 8x8: $\hat{B}'$. Em uma implementação, isso é executado por transformada inversa dos quatro sub-blocos DCT 4x4 de volta no domínio de pixel, e então aplicando um VC1-T 8x8 novo. Em uma implementação, e para reduzir a complexidade de computação, isso é executado no domínio DCT.

Por exemplo, faz-se $\hat{B}_1$, $\hat{B}_2$, $\hat{B}_3$, e $\hat{B}_4$ representar os quatro sub-blocos de baixa frequência 4x4 de B_1 , B_2 , B_3 , e B_4 , respectivamente; C_4 ser a matriz de transformada IDCT padrão 4x4; T_8 ser a matriz de transformada WMV inteira; e adicionalmente faz-se $T_8 = [T_L, T_R]$ onde T_L e T_R são matrizes 8x4. Nesse cenário, $\hat{B}$ é diretamente calculado a partir de $\hat{B}_1$, $\hat{B}_2$, $\hat{B}_3$, e $\hat{B}_4$ usando a seguinte equação:

$$\hat{B} = (T_L C'_4) \hat{B}_1 (T_L C'_4) + (T_L C'_4) \hat{B}_2 (T_R C'_4) \hat{B}_3 (T_L C')' + (T_R C'_4) \hat{B}_4 (T_R C'_4)'$$

Depois de alguma manipulação, $\hat{B}$ é mais eficientemente calculado como segue:

$$\hat{B} = (X + Y)C' + (X - Y)D'$$

onde

$$C = (T_L C'_4 + T_R C'_4) / 2$$

$$D = (T_L C'_4 - T_R C'_4) / 2$$

$$X = C(\hat{B}_1 + \hat{B}_3) + D(\hat{B}_1 - \hat{B}_3)$$

$$Y = C(\hat{B}_2 + \hat{B}_4) + D(\hat{B}_2 - \hat{B}_4)$$

Em uma implementação, ambos C e D da equação acima
 5 são pré-computados. Os resultados finais são normalizados
 com N_{88} .

A Fig. 8 mostra uma arquitetura exemplificada 800
 para um transcodificador de redução de resolução 2:1 numéri-
 ca no domínio DCT simplificado. Em uma implementação, o mó-
 dulo de transcodificação 408 da Fig. 4 implementa a arquite-
 10 tura exemplificada 800. Os comutadores nessa arquitetura têm
 a mesma funcionalidade daqueles da Fig. 6, como descrito a-
 cima com relação à TABELA 2. Com relação à Fig. 8, e uma im-
 plementação, os primeiros dois módulos (VLD MPEG-2 e quanti-
 15 zação inversa) são simplificados se comparado ao que é mos-
 trado na Fig. 6. Isso ocorre porque o módulo de transcodifi-
 cação 408 restaura somente a parte 4x4 esquerda superior do
 bloco 8x8.

Comparados a um transcodificador de baixo desvio
 20 convencional com compensação de erro de desvio em resolução
 reduzida, os transcodificadores das Figs. 6 e 8 não incluem
 um módulo de processamento de bloco misturado. Isso ocorre
 porque WMV suporta modo Intra Codificação para blocos 8x8 em
 um macrobloco inter codificado. Em outras palavras, um MB
 25 Intra na resolução original é mapeado em um intrabloco 8x8
 de um Inter MB na resolução reduzida. Em vista disso, a re-

gra de mapeamento de modo MB se torna muito simples, como mostrado imediatamente abaixo:

$$mode_new = \begin{cases} INTRA & \text{if all } mode_orig = INTRA \\ SKIP & \text{if all } mode_orig = SKIP \\ INTER & \text{otherwise} \end{cases}$$

As operações de processamento de bloco misturadas
5 existentes tipicamente exigem um laço de decodificação para reconstruir uma imagem de resolução completa. Portanto, a remoção de processamento de bloco misturado fornece substanciais economias de computação se comparado a sistemas convencionais.

10 A arquitetura de transcodificação de redução de resolução 2:1 no domínio DCT simplificada 800 é substancialmente livre de desvio para Quadros P. Esse é um resultado do modo de codificação de quatro MV. O único caso de erro de desvio, se comparado a uma arquitetura CPDT com filtragem de
15 redução, é o arredondamento de MVs de resolução de quarto para resolução de metade (que assegura $mv_{mp2} = mv_{vc1}$) e a propriedade não comutativa de MC e redução. Quaisquer tais erros restantes são desprezíveis devido à filtragem de redução passa-baixa (por exemplo, alcançado no domínio DCT ou no do-
20 mínio de pixel).

A Fig. 9 mostra uma operação de combinação exemplificada de quatro blocos DCT 4x4 em um bloco DCT 8x8 para meios de interlace para operações de transcodificação de redução de resolução espacial 2:1, de acordo com uma modalida-
25 de. A redução de 2:1 muda a resolução de um quadro original em dois, em ambas as direções horizontal e vertical. Em uma

implementação, esse processo de interlace é implementado pelo módulo de transcodificação 408 da Fig. 4. Mais particularmente, para conteúdo codificado de interlace, o sub-bloco 8x4 esquerdo superior em cada MB é reconstruído por decodificador MPEG-2 de atalho, ambos os campos são suavizados pelo filtro passa-baixa na direção vertical, então um campo é diminuído antes do processo de codificação WMV.

Compensação de Erro de MV

Embora WMV suporte quatro modos de codificação MV, ele é tipicamente somente pretendido para codificar Quadros P. Como um resultado, o sistema 400 (Fig. 4) implementa a arquitetura da Fig. 6 quando não há Quadros B no fluxo de entrada MPEG-2 ou os Quadros B são descartados durante o transcodificador em direção a uma resolução temporal mais baixa. Uma razão para isso é que WMV permite somente um MV por MB para Quadros B. Em tal cenário, o módulo de transcodificação 408 (Fig. 4) compõe um novo vetor de movimento a partir de quatro MVs associados com MBs na resolução original. Cada um dos métodos de composição MV anteriormente mencionados é compatível. Em uma implementação, o módulo de transcodificação 408 implementa filtragem mediana. Como descrito, MV incorreto levará a predição compensada por movimento errada. Para tornar as coisas piores, não importa como o erro de re-quantização é compensado, e não importa quão alta a taxa de bits é, resultados perfeitos são difíceis de obter se não re-fazendo a compensação de movimento baseada nos novos MVs. Portanto, fornece-se uma arquitetura que permite que tais erros de movimento sejam compensados.

Novamente, com relação à arquitetura da Fig. 3, a entrada para o codificador VC-1 para o quadro (i+1), que é assumido como sendo um quadro B, é derivada como segue:

$$e_{i+1} = D(\hat{r}_{i+1}) + D(MC_{mp2}(\hat{B}_i, MV_{mp2})) - MC_{vc1}(\tilde{b}_i, mv_{vc1}) \quad (9);$$

5 com a aproximação que

$$D(MC_{mp2}(\hat{B}_i, MV_{mp2})) = MC'_{mp2}(D(\hat{B}_i), D(MV_{mp2})) = MC'_{mp2}(\hat{b}_i, mv_{mp2}) \quad (10)$$

A Equação 9 é simplificada para:

$$e_{i+1} = D(\hat{r}_{i+1}) + MC'_{mp2}(\hat{b}_i, mv_{mp2}) - MC'_{vc1}(\tilde{b}_i, mv_{vc1}) \quad (11)$$

Em vista da Equação 11, a seguinte é obtida:

10

$$\begin{aligned} e_{i+1} &= D(\hat{r}_{i+1}) + MC'_{mp2}(\hat{b}_i, mv_{mp2}) - MC'_{vc1}(\tilde{b}_i, mv_{vc1}) \\ &= D(\hat{r}_{i+1}) + [MC'_{mp2}(\hat{b}_i, mv_{mp2}) - MC'_{vc1}(\hat{b}_i, mv_{vc1})] + MC'_{vc1}(\hat{b}_i, mv_{vc1}) - MC'_{vc1}(\tilde{b}_i, mv_{vc1}) \\ &= D(\hat{r}_{i+1}) + [MC'_{mp2}(\hat{b}_i, mv_{mp2}) - MC'_{vc1}(\hat{b}_i, mv_{vc1})] + MC'_{vc1}(\hat{b}_i - \tilde{b}_i, mv_{vc1}) \end{aligned} \quad (12)$$

Os dois termos entre colchetes na Equação 12 compensam os erros de movimento causados por MVs inconsistentes (isto é, mv_{mp2} é diferente de mv_{vc1}) ou causados por diferentes métodos de filtragem MC entre MPEG-2 e WMV. Os módulos correspondentes para esse propósito são destacados e agrupados em um bloco amarelo na Fig. 10.

A Fig. 10 mostra uma arquitetura de transcodificador de redução 2:1 simplificada exemplificada co compensação de desvio total, de acordo com uma modalidade. Em uma implementação, o módulo de transcodificação 408 da Fig. 4 implementa a arquitetura exemplificada da Fig. 10. Com relação à Equação 12, nota-se que $MC'_{mp2}(\hat{b}_i, mv_{mp2})$ é executado para todos os blocos 8x8 que correspondem aos Inter MBs originais, e $mv_{mp2} = MV_{mp2}/2$ com precisão de quarto de pixel. Nota-se que com relação ao módulo de compensação de erro de movimento, a

precisão de mvvc1 pode ir para o nível de quarto de pixel. O último termo na Equação 12 compensa o erro de re-quantização de quadros de referência. Como quadros B não são referência para outros quadros, eles são mais tolerantes a erros. Como
5 um resultado, um aplicativo pode seguramente desligar a compensação de erro para alcançar alta velocidade. Novamente, tal aproximação é pretendida para quadros B somente. Nota-se que MC para compensação de erro de movimento opera em armazenadores temporários de pixel reconstruído enquanto o MC
10 para compensação de erro de re-quantização opera em armazenador temporário de erro residual acumulado.

Como para o MC, a conversão Intra para Inter ou Inter para Intra pode ser aplicada. Isso ocorre porque o decodificador MPEG-2 reconstruiu o quadro B e os quadros de
15 referência. Nessa implementação, essa conversão é feita no módulo de processamento de bloco misturado na Fig. 10. Dois métodos de composição de modo são possíveis. E uma implementação, o modo dominante é selecionado como o modo composto. Por exemplo, se os modos dos quatro MBs na resolução original
20 são dois modos de predição bidirecionais, um modo de predição de retrocesso e um modo de predição de avanço, então o modo de predição bidirecional é selecionado como o modo para o MB na resolução reduzida. Em uma outra implementação, o modo que levará ao maior erro é selecionado. Em vista
25 desse exemplo, supõe-se que usar o modo de retrocesso causará o maior erro. Nesse cenário, o modo de retrocesso é escolhido tal que o erro pode ser compensado. Os resultados mostram que a última técnica oferece qualidade levemente melhor

se comparada à primeira técnica de seleção de modo.

Uma arquitetura exemplificada de acordo com a Equação 12 é mostrada na Fig. 10. Há quatro comutadores de nível de quadro para essa arquitetura, como mostrado na

5 TABELA 3.

TABELA 3

Comutadores de Nível de Quadro Exemplificados

S_{IP}	Nível de Quadro	Comutador a ser fechado para quadros I e P somente.
S_P	Nível de Quadro	Comutador a ser fechado para quadros P somente.
S_B	Nível de Quadro	Comutador a ser fechado para quadros B somente.
$S_{IP/B}$	Nível de Quadro	Comutador a ser fechado para quadros I e P somente se houver quadros B.

Os quatro comutadores de nível de quadro asseguram diferentes caminhos de codificação para diferentes tipos de quadro. Especificamente, a arquitetura não executa: acúmulo de erro residual para quadros B (S_{IP}), não executa compensação de erro de MV para quadros I e P (S_B), e não reconstrói quadros de referência se não houver quadros B a serem gerados ($S_{IP/B}$). Nota-se que o comutador de nível de quadro S_B pode ser transformado em comutador de nível de bloco já que o erro de MV necessita ser compensado somente quando os quatro MVs originais correspondentes são significativamente

10

15

inconsistentes.

Mais particularmente, o comutador S_{IP} é fechado somente para quadros I ou quadros P, o comutador S_P é fechado somente para quadros P, e o comutador S_B é fechado somente para quadros B. A arquitetura resultante não é tão complexa quanto o transcodificador no domínio de pixel em cascata de referência da Fig. 3. Uma razão para isso é que o processo de redução no domínio de pixel explícito é evitado. Ao invés, a redução no domínio de pixel é implicitamente alcançada no domínio DCT simplesmente descartando os altos coeficientes DCT. Essa arquitetura tem excelente escalabilidade de complexidade alcançada utilizando-se vários comutadores, como descrito acima com relação à TABELA 2.

Para aplicações que demandam velocidade de transcodificação ultra-rápida, a arquitetura da Fig. 10 pode ser configurada em um laço aberto desligando todos os comutadores. Essa arquitetura de laço aberto pode ser adicionalmente otimizada combinando-se o processo de desquantização de MPEG-2 e o processo de re-quantização de WMV. O módulo de digitalização em zigue-zague inverso (dentro de VLD) de MPEG-2 pode também ser combinado com o no codificador WMV.

Componentes de Crominância

Com relação a componentes de crominância em MPEG-2 e em WMV, o MV e o modo de codificação de componentes de crominância (UV) são derivados daqueles do componente de luminância (Y). Se todos os quatro MBs na resolução original que correspondem ao MB na resolução reduzida têm modo de codificação consistente (isto é, todos Intercodificados ou to-

dos Intracodificados), não há problema. Entretanto, se não é o caso, problemas resultam devido a diferentes regras de derivação de MPEG-2 e WMV. Em MPEG-2, os blocos UV são Inter-

codificados quando o MB é codificado com Intermodo. Entretanto, em WMV, os blocos UV são Inter

codificados somente quando o MB é codificado com Inter modo e há menos do que três blocos Y 8x8 intracodificados. Esse problema existe para ambos os quadros P e os quadros B. O módulo de transcodificação 408 da Fig. 4 aborda esses problemas como segue:

10 - Conversão Inter para Intra: Quando o MB intercodificado tem três blocos Y 8x8 intracodificados (é impossível para um MB intercodificado ter todos os quatro blocos Y 8x8 intracodificados), os blocos UV são intracodificados. Nesse caso, um MB na resolução original é intracodificado

15 junto com blocos UV correspondentes. Esses blocos UV serão convertidos de Intermodo em Intramodo. Como o Sistema Visual Humano (HVS) é menos sensível aos sinais de crominância, o módulo de transcodificação 408 utiliza uma técnica de ocultação espacial para converter os blocos UB 8x8 de Intermodo

20 para Intramodo. Em uma implementação, a distância DC é utilizada como um indicador para determina a direção da ocultação. A ocultação é alcançada via uma simples cópia ou qualquer outro método de interpolação.

 - Conversão Intra para Inter: Quando um MB inter-

25 codificado tem um ou dois blocos Y 8x8 intracodificados, o módulo de transcodificação 408 intercodifica os blocos UV. Nesse cenário, há um ou dois MBs intracodificados entre os quatro MBs correspondentes na resolução original. Esses blo-

cos UV são convertidos de intramodo para intermodo. Nessa implementação, o módulo de transcodificação 408 utiliza uma técnica de ocultação temporal chamada o método zero-out para manipular esses blocos, e desse modo, evitar o laço de deco-
5 dificação.

Usando operações de ocultação de erro para manipular a conversão de modo para componente de crominância, o erro introduzido em um quadro atual é desprezível e pode ser ignorado, embora possa causar desvio de cor em quadros sub-
10 seqüentes. O desvio para o componente de crominância é tipicamente causado por movimento incorreto. Para abordar isso e aperfeiçoar a qualidade, em uma implementação, o módulo de transcodificação 408 usa compensação baseada em reconstrução para o componente de crominância (isto é, sempre aplicando o
15 módulo amarelo claro para o componente de crominância).

Controle de Taxa

A Fig. 11 mostra um modelo de verificador de armazenador temporário virtual exemplificado (VBV) para um decodificador. Um decodificador baseado no modelo VBV da Fig. 11
20 tipicamente verificará um fluxo de bits MPEG-2 existente. Nessa implementação, se a taxa de vídeo é diminuída proporcional à taxa de entrada, então o fluxo de bits WMB transcodificado automaticamente satisfará às exigências de VBV. Em vista disso, a arquitetura eficiente de transcodificação de
25 vídeo digital dessa especificação torna o tamanho de quadro codificado proporcional ao tamanho do quadro de entrada para todos os quadros. Essas novas arquiteturas continuamente compensam diferenças acumuladas entre o tamanho de quadro

alvo e o tamanho de quadro resultante real, e obtêm, via treinamento, uma regra de mapeamento de etapa de quantização linear (QP) para diferentes faixas de taxa de bits.

Para taxa de bits alta, há uma fórmula aproximada entre os bits de codificação (B) e a etapa de quantização (QP) que é também usada no método de controle de taxa TM-5 MPEG-2.

$$B = S \cdot \frac{X}{QP}$$

(13)

onde S é a complexidade de quadro, X é parâmetros modelo. Assume-se que a complexidade de um quadro permanece a mesma para diferentes codecs:

$$QP_{vc1} = \left(\frac{X_{vc1}}{X_{mp2}} \right) \cdot \left(\frac{B_{mp2}}{B_{vc1}} \right) \cdot QP_{mp2} = k \cdot \left(\frac{B_{mp2}}{B_{vc1}} \right) \cdot QP_{mp2}$$

onde QP_{vc1} é o valor QP usado na re-quantização WMV, QP_{mp2} é o valor QP de quantização MPEG-2, e k é o parâmetro modelo relacionado à taxa de bits alvo. Em uma implementação, o seguinte modelo linear é utilizado:

$$QP_{vc1} / QP_{mp2} = k \cdot (B_{mp2} / B_{vc1}) + t$$

(14)

Os valores do parâmetro k e t para os casos de taxa de bits baixa, média e alta são listados na TABELA 4 usando o método de regressão linear.

TABELA 4

VALORES DE PARÂMETRO EXEMPLIFICADOS PARA
METODOLOGIA DE REGRESSÃO LINEAR

Tipo de Quadro	Quadro I		Quadro P		Quadro B	
Parâmetros	k	t	k	t	k	t
Baixa (< 1Mbps)	0,61286 1	- 0,19495 4	0,01608 1	3,1285 61	0,07603 7	2,2648 25
Média (< 3Mbps)	0,31431 1	0,07049 4	0,04114 0	1,4006 47	0,20729 2	0,5459 77
Alta	0,68240 9	- 0,24812 0	0,05786 9	1,1159 30	0,19902 4	0,4415 18

Um algoritmo de controle de taxa detalhado exemplificado baseado na Equação 14 é mostrado na TABELA 5, onde os significados dos vários símbolos no algoritmo apresentado na TABELA 5 são definidos na seguinte TABELA 6.

5

TABELA 5

ALGORITMO DE CONTROLE DE TAXA EXEMPLIFICADO

Inicializar SumD = 0;

Enquanto (fluxo MPEG-2 não terminou)

{

10

Etapa 1: Decodificar um quadro MPEG2 e obter B_{mp2} e QP_{mp2} ;

Etapa 2:

$$B_{pred_vc1} = B_{mp2} \cdot \frac{R_{vc1}}{R_{mp2}}$$

$$B_{vc1} = B_{pred_vc1} + SumD$$

Se $(B_{vc1} < 0)$ então $B_{vc1} = 1$;

$$QP_{vc1} = \left(k \cdot \frac{B_{mp2}}{B_{vc1}} + t \right) \cdot QP_{mp2};$$

Arredondar e Cortar QP_{vc1} para $[1, 31]$;

Etapa 3: Codificar esse quadro em quadro WMV usando QP_{vc1} ;

5 Etapa 4: Obter o tamanho de quadro WMV codificado real B_{real_vc1} ;

Atualizar SumD: $SumD = SumD + B_{pred_vc1} - B_{real_vc1}$;

}

TABELA 6

10 DEFINIÇÕES DE SÍMBOLOS USADOS NO ALGORITMO DA

TABELA 5

B_{mp2}	Tamanho de quadro MPEG-2
R_{mp2}	Taxa de bits de fluxo MPEG-2;
R_{vc1}	Taxa de bits de fluxo WMV alvo;
B_{pred_vc1}	Tamanho de quadro WMV previsto pela razão da taxa de bits;
B_{vc1}	Tamanho de quadro WMV esperado para codificar (nova taxa de bits);
B_{real_vc1}	Tamanho de quadro WMV codificado real;
SumD	Diferenças acumuladas entre o tamanho de quadro previsto e real do início.

Mudança de Resolução Arbitrária

A conversão de conteúdos de resolução HD para resolução SD, por exemplo, para suportar receptores/reprodutores SD de legado, é útil. Resoluções típicas de formato HS são 1920x1080i e 1280x720p enquanto aquelas para SD são 720x480i, 720x480p para NTSC. As razões de redução horizontal e vertical de 1920x1080i para 720x480i são 8/3 e 9/4, respectivamente. Para manter a taxa de definição, a razão de redução final é escolhida como sendo 8/3 e o tamanho da imagem resultante é 720x404. Similarmente, para 1280x720p para 720x480p, a razão de redução é escolhida como sendo 16/9 e o tamanho de imagem resultante é 720x404. Tarjas pretas são inseridas para fazer uma imagem completa 720x480 pelo decodificador/reprodutor (ao invés de ser preenchida no fluxo de bits).

De acordo com a teoria de processamento de sinal digital, uma metodologia de redução substancialmente ótima para uma razão de redução m/n seria primeiro superamostrar o sinal em n vezes (isto é, inserir $n-1$ zeros entre cada amostra original), aplicar um filtro passa-baixa (por exemplo, uma função sincronização com muitas derivações), e então decimar o sinal resultante em m vezes. Executando essas operações, qualquer sobreposição de espectro introduzido pela redução seria suprimida ao máximo. Entretanto, esse processo seria também computacionalmente muito dispendioso, e difícil de implementar em tempo real porque o sinal de entrada é de alta definição. Para reduzir essa complexidade computacional, uma nova estratégia de redução em dois estágios é im-

plementada.

A Fig. 12 mostra um transcodificador com redução de resolução espacial arbitrária, de acordo com uma modalidade. Em uma implementação, o módulo de transcodificação 408 da Fig. 4 implementa a arquitetura da Fig. 12. Em uma imple-
 5 mentação, o transcodificador de redução arbitrária é um transcodificador não integrado, tal como na Fig. 12. Em uma outra implementação, as seguintes operações de transcodificação de redução arbitrária, que são descritas abaixo com
 10 relação à Fig. 12, são implementadas em um transcodificador integrado tal como aquele mostrado nas Figs. 5, 6, 8 e/ou 10.

Com relação à Fig. 12, o sistema 1200 implementa operações de redução em dois estágios para alcançar qualquer
 15 alvo de redução arbitrária. Resultados da redução em primeiro estágio são embutidos no laço de decodificação. Isso reduz a complexidade das operações de decodificação. Por exemplo, para alcançar uma razão de redução $8/3$, operações de redução são primeiro implementadas para reduzir em $2/1$. Os
 20 resultados dessa redução em primeiro estágio são inseridos no laço de decodificação, onde a redução em segundo estágio é executada no domínio espacial. Nesse exemplo, as operações de redução em segundo estágio reduzem em $4/3$ para alcançar uma razão de redução de $8/3$. Em um outro exemplo, uma razão
 25 de $16/9$ é alcançada pelo sistema 1200 aplicando-se a redução de $4/3$ duas vezes (em dois estágios). Essa metodologia de redução em dois estágios utiliza a estratégia de redução no domínio DCT discutida anteriormente, e então embute comple-

tamente os resultados de redução de primeiro estágio no laço de decodificação. Como a resolução é significativamente reduzida depois da redução em primeiro estágio, pode-se continuar a aplicar o método de redução ótima no domínio de pixel.

Com relação à Fig. 12, nota-se que múltiplos MVs (entre $\left\lfloor \frac{m}{n} \right\rfloor \times \left\lfloor \frac{m}{n} \right\rfloor$ e $\left\lceil \frac{m}{n} \right\rceil \times \left\lceil \frac{m}{n} \right\rceil$)

estão associados com um novo MB (os módulos de dimensionamento e filtragem MV).

10 Procedimento Exemplificado

A Fig. 13 ilustra um procedimento 1300 para eficiente transcodificação de vídeo digital, de acordo com uma modalidade. Em uma implementação, o módulo de transcodificação 408 da Fig. 4 implementa as operações do procedimento 1300. Com relação à Fig. 13, no bloco 1302, o procedimento recebe um fluxo de bits codificado (por exemplo, meios codificados 412 da Fig. 4). No bloco 1304, o procedimento parcialmente decodifica o fluxo de bits codificado de acordo com um primeiro conjunto de técnicas de compressão associadas com um primeiro formato de dados de mídia (por exemplo, MPEG-2, MPEG-4, etc.). As operações de decodificação parcial geram um fluxo de dados intermediário. O transcodificador integrado não executa decodificação completa. Por exemplo, em casos onde o MC do decodificador MPEG_2 "conceitual" é combinado com aquele do codificador WMV, é difícil descrever as operações de decodificação à medida que executando a decodificação MPEG-2. No bloco 1306, se a redução do fluxo de dados intermediário é desejada, o procedimento reduz dados

associados com o fluxo de bits codificado em um primeiro estágio de redução. O primeiro estágio de redução é implementado no domínio DCT de um laço de decodificação. No bloco 1308, se redução em dois estágios é desejado, o procedimento
5 adicionalmente reduz no domínio espacial os dados que foram reduzidos no domínio DCT (ver bloco 1306).

No bloco 1310, os dados decodificados de acordo com o primeiro conjunto de técnicas de compressão são codificados com um segundo conjunto de técnicas de compressão.
10 Em uma implementação, o procedimento 1300 é implementado em uma arquitetura de transcodificação não integrada, tal como aquela mostrada e descrita com relação às Figs. 12 e 14. Nessa implementação, o segundo conjunto de técnicas de compressão é o mesmo do primeiro conjunto de técnicas de compressão.
15 Em uma outra implementação, o procedimento 1300 é implementado em uma arquitetura de transcodificação integrada, tal como aquela mostrada e descrita com relação às Figs. 5-11, e 14. Nessa outra implementação, o segundo conjunto de técnicas de compressão não é o mesmo do primeiro conjunto de técnicas de compressão.
20 técnicas de compressão. Por exemplo, em uma implementação, o primeiro conjunto de técnicas de compressão está associado com MPEG-2, e o segundo conjunto de técnicas de compressão está associado com WMV.

Um Ambiente Operacional Exemplificado

25 A Fig. 14 ilustra um exemplo de um ambiente de computação adequado no qual eficiente transcodificação de vídeo digital pode ser completa ou parcialmente implementada. O ambiente de computação exemplificado 1400 é somente um

exemplo de um ambiente de computação adequado para o sistema exemplificado 400 da Fig. 4, e não pretende sugerir qualquer limitação como ao escopo de uso ou funcionalidade de sistemas e métodos descritos aqui. Nem deveria o ambiente de computação 1400 ser interpretado como tendo qualquer dependência ou exigência relativa a qualquer um dos componentes ou combinação deles ilustrada no ambiente de computação 1400.

Os métodos e sistemas descritos aqui são operacionais com numerosos outros sistemas de computação de propósito geral ou de propósito especial, ambientes ou configurações. Exemplos de sistemas de computação, ambientes e/ou configurações bem conhecidos que podem ser adequados para uso incluem, mas não estão limitados a computadores pessoais, computadores servidores, sistemas multiprocessadores, sistemas baseados em microprocessador, PCs de rede, minicomputadores, computadores de grande porte, ambientes de computação distribuídos que incluem quaisquer dos sistemas ou dispositivos acima, e assim por diante. Versões de subconjunto ou compactas da estrutura podem também ser implementadas em clientes de recursos limitados, tal como computadores portáteis, ou outros dispositivos de computação. A invenção é praticada em um ambiente de computação em rede onde tarefas são executadas por dispositivos de processamento remoto que são ligados através de uma rede de comunicação.

Com relação à Fig. 14, um sistema exemplificado fornecendo eficiente arquitetura de transcodificação de vídeo digital inclui um dispositivo de computação de propósito

geral na forma de um computador 1410 implementando, por exemplo, operações iniciadoras associadas com o dispositivo de computação 102 da Fig. 1. Componentes do computador 1410 podem incluir, mas não estão limitados a, unidade(s) de processamento 1418, uma memória de sistema 1430, e um barramento de sistema 1421 que acopla vários componentes de sistema, incluindo a memória de sistema, à unidade de processamento 120. O barramento de sistema 1421 pode ser qualquer um dos vários tipos de estruturas de barramentos incluindo um barramento de memória ou controlador de memória, um barramento periférico, e um barramento local usando qualquer uma de uma variedade de arquiteturas de barramento. A título de exemplo, e não limitação, tais arquiteturas podem incluir barramento de Arquitetura Padrão de Indústria (ISA), barramento de Arquitetura de Micro Canal (MCA), barramento ISA Aperfeiçoado (EISA), barramento local de Associação de Padrões Eletrônicos de Vídeo (VESA), e barramento de Componentes Periféricos Interconectados (PCI), também conhecido como barramento Mezzanine.

Um computador 1410 tipicamente inclui uma variedade de meios legíveis por computador. Estes podem ser quaisquer meios disponíveis que podem ser acessados pelo computador 1410 e incluem ambos meios voláteis e não voláteis, removíveis e não removíveis. A título de exemplo, e não limitação, meios legíveis por computador podem compreender meios de armazenamento por computador e meios de comunicação. Os meios de armazenamento por computador incluem ambos meios voláteis e não voláteis, removíveis e não removíveis imple-

mentados em qualquer método ou tecnologia para armazenamento de informação tal como instruções legíveis por computador, estruturas de dados, módulos de programa ou outros dados. Os meios de armazenamento por computador incluem, mas não estão
5 limitados a, RAM, ROM, EEPROM, memória rápida ou outra tecnologia de memória, CD-ROM, discos versáteis digitais (DVD) ou outro armazenamento em disco óptico, cassetes magnéticos, fita magnética, armazenamento em disco magnético ou outros dispositivos de armazenamento magnéticos, ou qualquer outro
10 meio que pode ser usado para armazenar a informação desejada e que pode ser acessado pelo computador 1410.

Os meios de comunicação tipicamente incorporam instruções legíveis por computador, estruturas de dados, módulos de programa ou outros dados em um sinal modulado de
15 dados tal como uma onda portadora ou outro mecanismo de transporte, e incluem quaisquer meios de entrega de informação. O termo "sinal modulado de dados" significa um sinal que tem uma ou mais de suas características ajustadas ou alteradas de tal maneira a codificar informação no sinal. A
20 título de exemplo, e não limitação, os meios de comunicação incluem meios por fios tais como uma rede por fios ou conexão direta por fios, e meios sem fio tais como acústicos, RF, infravermelhos e outros meios sem fio. Combinações de qualquer um dos acima deveriam ser também incluídas no escopo dos meios legíveis por computador.
25

A memória de sistema 1430 inclui meios de armazenamento por computador na forma de memória volátil e/ou não volátil, tal como memória somente de leitura (ROM) 1431 e

memória de acesso aleatório (RAM) 1432. Um sistema de entrada/saída básico (BIOS) 1433, contendo as rotinas básicas que ajudam a transferir informação entre elementos no computador 1410, tal como durante a inicialização, é tipicamente armazenado na ROM 1431. A RAM 1432 tipicamente contém dados e/ou módulos de programa que são imediatamente acessíveis e/ou estão presentemente sendo operados pela unidade de processamento 1418. A título de exemplo, e não limitação, a FIG. 14 ilustra o sistema operacional 1434, programas de aplicação 1435, outros módulos de programa 1436, e dados de programa 1437.

O computador 1410 pode também incluir outros meios de armazenamento por computador removíveis/não removíveis, voláteis/não voláteis. A título de exemplo somente, a FIG. 14 ilustra uma unidade de disco rígido 1441 que lê a partir de e escreve em meios magnéticos não removíveis e não voláteis, uma unidade de disco magnético 1451 que lê a partir de e escreve em discos magnéticos removíveis e não voláteis 1452, e uma unidade de disco óptico 1455 que lê a partir de e escreve em um disco óptico removível e não volátil 1456, tal como um CD-ROM ou outro meio óptico. Outros meios de armazenamento por computador removíveis/não removíveis, voláteis/não voláteis que podem ser usados na modalidade operacional exemplificada incluem, mas não estão limitados a, fitas cassetes magnéticas, cartões de memória rápida, discos versáteis digitais (DVDs), fita de vídeo digital, RAM de estado sólido, ROM de estado sólido, e seus similares. A unidade de disco rígido 1441 é tipicamente conectada ao barra-

mento de sistema 1421 através de uma interface de memória não removível tal como a interface 1440, e a unidade de disco magnético 1451 e a unidade de disco óptico 1455 são tipicamente conectadas ao barramento de sistema 1421 por uma interface de memória removível, tal como a interface 1450.

As unidades e seus meios de armazenamento por computador associados discutidos acima e ilustrados na FIG. 14 fornecem armazenamento de instruções legíveis por computador, estruturas de dados, módulos de programa e outros dados para o computador 1410. Na FIG. 14, por exemplo, a unidade de disco rígido 1441 é ilustrada como armazenando o sistema operacional 1444, programas de aplicação 1445, e módulos de programa 1446, e dados de programa 1447. Nota-se que esses componentes podem ou ser os mesmos ou diferentes do sistema operacional 1434, programas de aplicação 1435, outros módulos de programa 1436, e dados de programa 1437. Ao sistema operacional 1444, aos programas de aplicação 1445, a outros programas 1446 e aos dados de programa 1447 são atribuídos números diferentes aqui para ilustrar que, no mínimo, eles são cópias diferentes.

Um usuário pode inserir comandos e informação no computador 1410 através de dispositivos de entrada, tais como um teclado 1462 e um dispositivo de apontamento 1461, comumente referido como um mouse, mouse estacionário ("trackball") ou mesa sensível ao toque. Outros dispositivos de entrada (não mostrados) podem incluir um microfone, um comando de jogos, caneta gráfica ou mesa gráfica, antena de satélite, digitalizador, etc. Esses e outros dispositivos de en-

trada são freqüentemente conectados à unidade de processamento 1418 através de uma interface de entrada de usuário 1460 que está acoplada ao barramento de sistema 1421, mas pode estar conectada por outra interface e estruturas de barramento, tais como uma porta paralela, porta de jogos ou um barramento serial universal (USB). Nessa implementação, um monitor 1491 ou outro tipo de dispositivo de interface de usuário está também conectado ao barramento de sistema 1421 via uma interface, por exemplo, tal como uma interface de vídeo 1490.

O computador 1410 opera em um ambiente de rede usando conexões lógicas a um ou mais computadores remotos, tal como um computador remoto 1480. Em uma implementação, o computador remoto 1480 representa o dispositivo de computação 106 de um responder, como mostrado na Fig. 1. O computador remoto 1480 pode ser um computador pessoal, um servidor, um roteador, um PC de rede, um dispositivo não hierárquico ou outro nó de rede comum, e como uma função de sua implementação particular, pode incluir muitos ou todos dos elementos descritos acima em relação ao computador 1410, embora somente um dispositivo de armazenamento em memória 1481 foi ilustrado na FIG. 14. As conexões lógicas representadas na FIG. 14 incluem uma rede de área local (LAN) 1471 e uma rede de área ampla (WAN) 1473, mas podem também incluir outras redes. Tais ambientes de rede são comuns em escritórios, redes de computador de grandes empresas, intranets e a Internet.

Quando usado em um ambiente de rede LAN, o compu-

tador 1410 é conectado à LAN 1471 através de uma interface de rede ou adaptador 1470. Quando usado em um ambiente de rede WAN, o computador 1410 tipicamente inclui um modem 1472 ou outro dispositivo para estabelecer comunicações pela WAN 5 1473, tal como a Internet. O modem 1472, que pode ser interno ou externo, pode estar conectado ao barramento de sistema 1421 via a interface de entrada de usuário 1460, ou outro mecanismo apropriado. Em um ambiente de rede, módulos de programa representados em relação ao computador 1410, ou 10 partes desse, podem ser armazenados no dispositivo de armazenamento em memória remota. A título de exemplo, e não limitação, a FIG. 14 ilustra programas de aplicação remota 1485 como residindo no dispositivo de memória 1481. As conexões de rede mostradas são exemplificadas e outros meios de 15 estabelecer uma ligação de comunicação entre os computadores podem ser usados.

Conclusão

Embora as seções acima descrevem decodificadores de redução de mudança de resolução arbitrária em linguagem 20 específica a características estruturais e/ou operações metodológicas ou ações, as implementações definidas nas reivindicações em anexo não são necessariamente limitadas às características ou ações específicas descritas. De preferência, características e operações específicas do decodificador de redução de mudança de resolução arbitrária são des- 25 critas como formas exemplificadas de implementar o assunto reivindicado.

Por exemplo, em uma implementação, os sistemas e

metodologias de transcodificação rápidos e de alta qualidade, incluindo transcodificação, redução de tamanho arbitrário, e redução de taxa são usados para transcodificação MPEG-2 e MPEG-4 e transcodificação MPEG-4 para WMV. Por exemplo, o transcodificador em domínio DCT de laço fechado simplificado na Fig. 6 pode ser usado para transcodificar MPEG-4 para WMV. Uma diferença entre MPEG-2 (IS-13818 Parte 2) é que MPEG-2 somente utiliza precisão de MPV de meio elemento de pixel (pel) e interpolação bilinear em MC; há tal mesmo modo (meio pel bilinear) em WMV. Entretanto, MPEG-4 suporta ambas a precisão de MV de meio pel e a de um quarto de pel, bem como interpolação para posições de quarto de pel (diferente daquela em WMV). Para abordar essa diferença, quando $\frac{1}{2}$ pel MV é usado por vídeo MPEG-4, então o processo de transcodificação é o mesmo da transcodificação MPEG-2 para WMV, como descrito acima. Adicionalmente, quando $\frac{1}{4}$ pel MV está contido em vídeo MPEG-4, então erro é introduzido devido a métodos de interpolação diferentes em MC como descrito acima com relação às Fig. 6. Adicionalmente, o transcodificador de redução 2:1 simplificado com compensação de desvio completa descrito acima com relação à Fig. 10 é aplicável à transcodificação de redução 2:1 MPEG-4 a WMV independente de mudança. Além disso, a transcodificação de alta qualidade, incluindo operações de redução de taxa e de transcodificação de redução arbitrária descritas acima da Fig. 12 são efetivas para transcodificação MPEG-4 a WMV.

REIVINDICAÇÕES

1. Método implementado por computador, **CARACTERIZADO** pelo fato de que compreende:

receber um fluxo de bits codificado; e

5 reduzir o fluxo de bits codificado em um laço de decodificação no domínio DCT para gerar dados reduzidos.

2. Método implementado por computador, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que um decodificador no Domínio de Transformada Discreta de Cosseno
10 em Cascata (CDDT) implementa a redução.

3. Método implementado por computador, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que reduzir o fluxo de bits codificado adicionalmente compreende converter dados associados com o fluxo de bits codificado de
15 uma resolução em alta definição para uma resolução em definição padrão.

4. Método implementado por computador, de acordo com a reivindicação 1, **CARACTERIZADO** adicionalmente pelo fato de que compreende reduzir a resolução dos dados reduzidos
20 em um domínio de pixel.

5. Método implementado por computador, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que a redução é redução em dois estágios que compreende redução em primeiro e segundo estágio, a redução no primeiro estágio
25 resulta em dados reduzidos 2:1 ou 4:3, a redução no segundo estágio resulta em uma redução 4:3 adicional ou uma redução passa tudo nos dados baseada em uma razão de redução de resolução alvo.

6. Método implementado por computador, de acordo com a reivindicação 1, **CARACTERIZADO** pelo fato de que a redução é em dois estágios executada em um transcodificador integrado, este parcialmente decodifica o fluxo de bits codificado baseado em um primeiro conjunto de técnicas de compressão para gerar um fluxo de dados intermediário, o transcodificador integrado codifica o fluxo de dados intermediário de acordo com um segundo conjunto de técnicas de compressão que são diferentes das do primeiro conjunto de técnicas de compressão.

7. Método implementado por computador, de acordo com a reivindicação 6, **CARACTERIZADO** pelo fato de que:

o primeiro conjunto de técnicas de compressão está associado com MPEG-2 e o segundo conjunto de técnicas de compressão está associado com WMV; ou

o primeiro conjunto de técnicas de compressão está associado com MPEG-2 e o segundo conjunto de técnicas de compressão está associado com MPEG-4.

8. Método implementado por computador, **CARACTERIZADO** pelo fato de que compreende:

receber um fluxo de bits codificado; e

parcialmente decodificar o fluxo de bits codificado, este compreende:

reduzir dados associados com o fluxo de bits codificado em um laço de decodificação no domínio DCT para obter dados reduzidos; e

codificar os dados em um formato de mídia alvo.

9. Método implementado por computador, de acordo

com a reivindicação 8, **CARACTERIZADO** pelo fato de que um decodificador no Domínio de Transformada Discreta de Cosseno em Cascata (CDDT) implementa a redução.

10 10. Método implementado por computador, de acordo com a reivindicação 8, **CARACTERIZADO** pelo fato de que reduzir o fluxo de bits codificado adicionalmente compreende converter dados associados com o fluxo de bits codificado de uma resolução em alta definição em uma resolução em definição padrão.

10 11. Método implementado por computador, de acordo com a reivindicação 8, **CARACTERIZADO** adicionalmente pelo fato de que compreende, antes da codificação, reduzir a resolução dos dados reduzidos em um domínio de pixel.

15 12. Método implementado por computador, de acordo com a reivindicação 8, **CARACTERIZADO** pelo fato de que a redução é em dois estágios resultando em redução no primeiro estágio de 2:1 ou 4:3, e redução em segundo estágio adicional de 4:3 ou uma determinação passa tudo baseada em uma razão de redução alvo.

20 13. Método implementado por computador, de acordo com a reivindicação 8, **CARACTERIZADO** pelo fato de que um transcodificador integrado implementa a redução, este parcialmente decodifica o fluxo de bits codificado de acordo com um primeiro conjunto de técnicas de compressão para gerar um
25 fluxo de dados intermediário, o transcodificador integrado codifica o fluxo de dados intermediário de acordo com um segundo conjunto de técnicas de compressão que é diferente do primeiro conjunto de técnicas de compressão.

14. Método implementado por computador, de acordo com a reivindicação 13, **CARACTERIZADO** pelo fato de que:

o primeiro conjunto de técnicas de compressão está associado com MPEG-2 e o segundo conjunto de técnicas de compressão está associado com WMV; ou

o primeiro conjunto de técnicas de compressão está associado com MPEG-2 e o segundo conjunto de técnicas de compressão está associado com MPEG-4.

15. Método implementado por computador, **CARACTERIZADO** pelo fato de que compreende:

transcodificar o fluxo de bits codificado usando um primeiro conjunto de técnicas de compressão tal que dados associados com o fluxo de bits codificado em um laço de decodificação no domínio DCT para obter dados reduzidos; e

codificar os dados reduzidos em um formato de mídia alvo baseado no primeiro conjunto de técnicas de compressão ou no segundo conjunto de técnicas de compressão que é diferente do primeiro conjunto de técnicas de compressão.

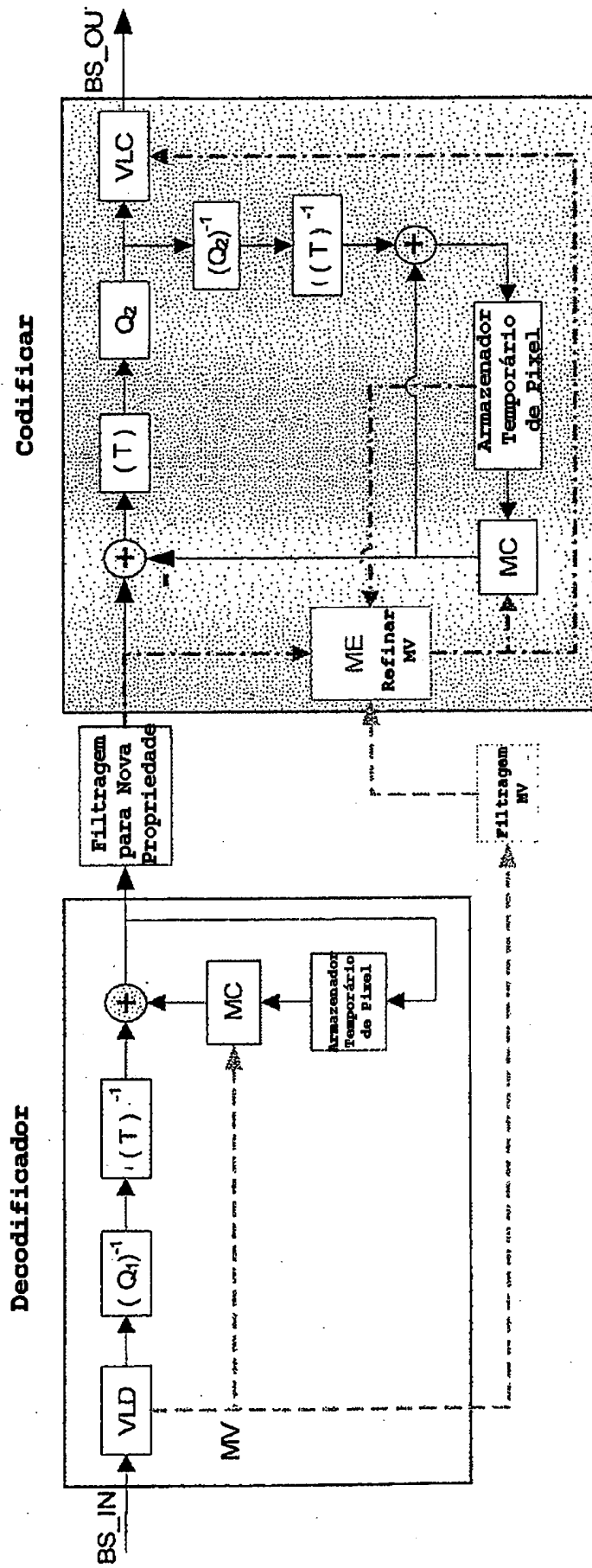
16. Método implementado por computador, de acordo com a reivindicação 15, **CARACTERIZADO** pelo fato de que um decodificador no Domínio de Transformada Discreta de Cosseno em Cascata (CDDT) implementa a redução.

17. Método implementado por computador, de acordo com a reivindicação 15, **CARACTERIZADO** pelo fato de que reduzir o fluxo de bits codificado adicionalmente compreende converter dados associados com o fluxo de bits codificado de uma resolução em alta definição em uma resolução em definição padrão.

18. Método implementado por computador, de acordo com a reivindicação 15, **CARACTERIZADO** adicionalmente pelo fato de que compreende, antes da codificação, reduzir a resolução dos dados reduzidos em um domínio de pixel fora de um domínio DCT.

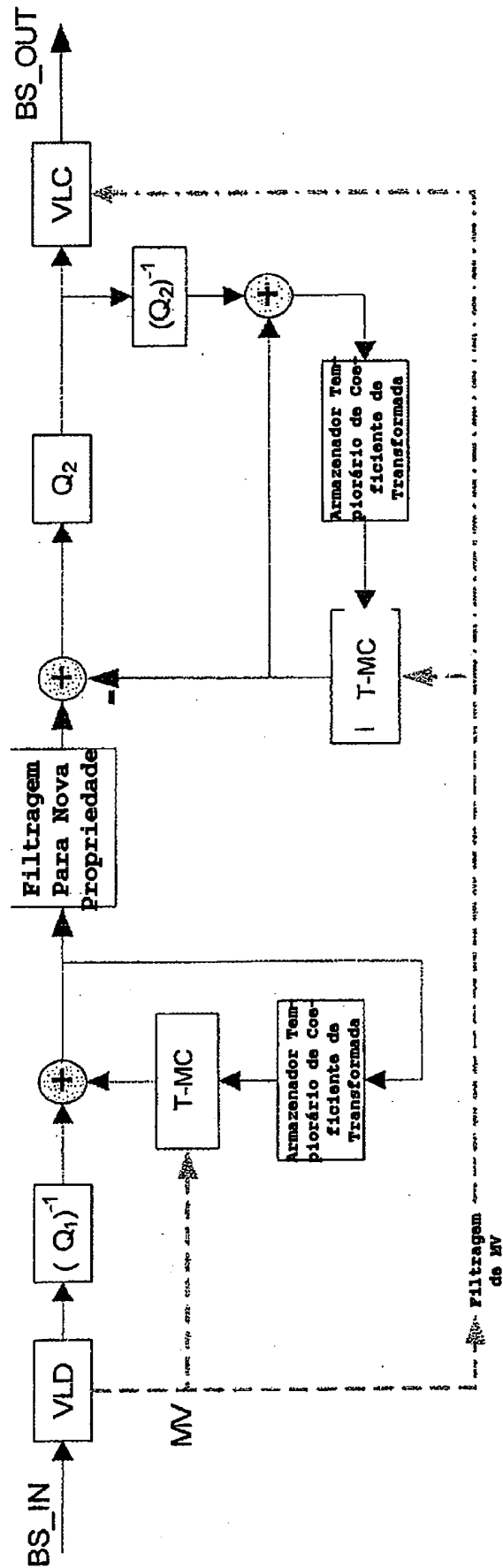
19. Método implementado por computador, de acordo com a reivindicação 15, **CARACTERIZADO** pelo fato de que a redução é em dois estágios resultando em redução no primeiro estágio de 2:1 ou 4:3, e redução em segundo estágio adicional de 4:3 ou uma determinação passa tudo baseada em uma razão de redução alvo.

20. Método implementado por computador, de acordo com a reivindicação 15, **CARACTERIZADO** pelo fato de que um transcodificador integrado implementa a redução, este parcialmente decodifica o fluxo de bits codificado de acordo com um primeiro conjunto de técnicas de compressão para gerar um fluxo de dados intermediário, o transcodificador integrado codifica o fluxo de dados intermediário de acordo com um segundo conjunto de técnicas de compressão que é diferente do primeiro conjunto de técnicas de compressão.



Técnica
Anterior

Fig. 1



Técnica
Anterior
Fig. 2

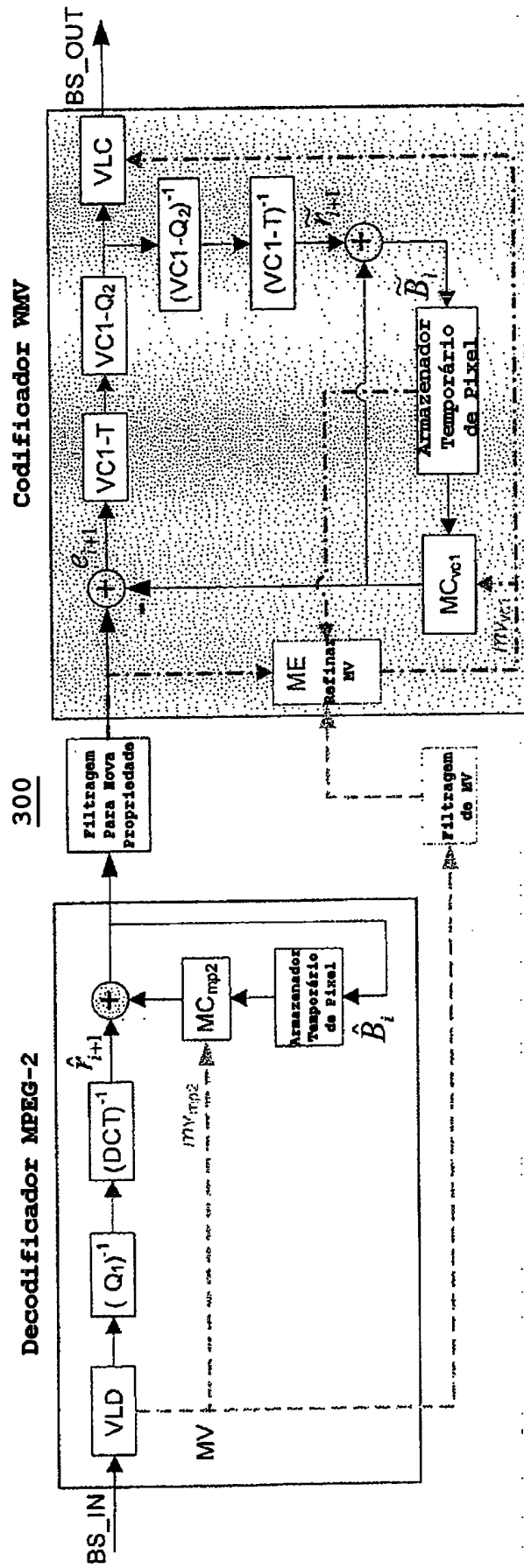


Fig. 3

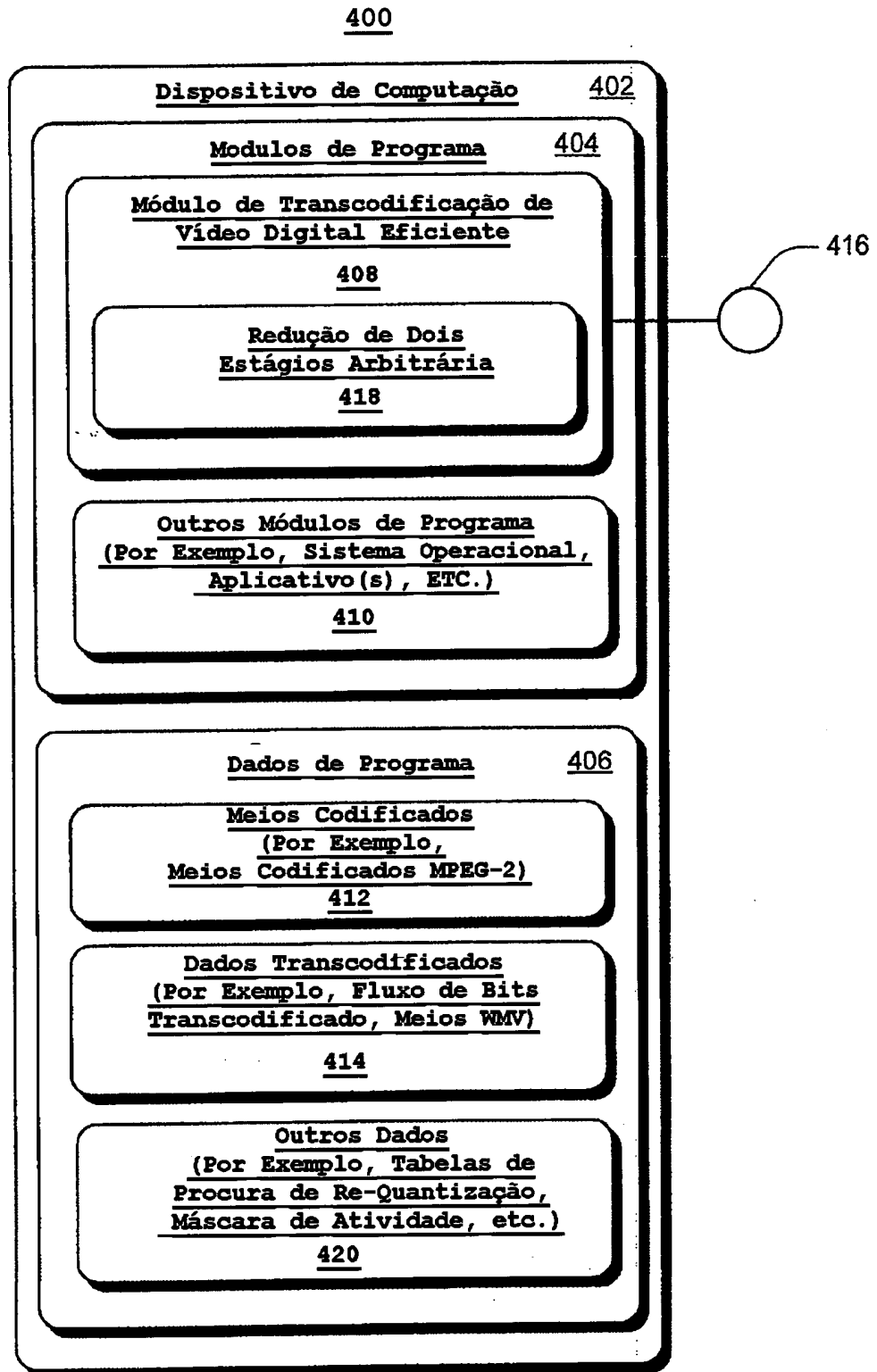


Fig. 4

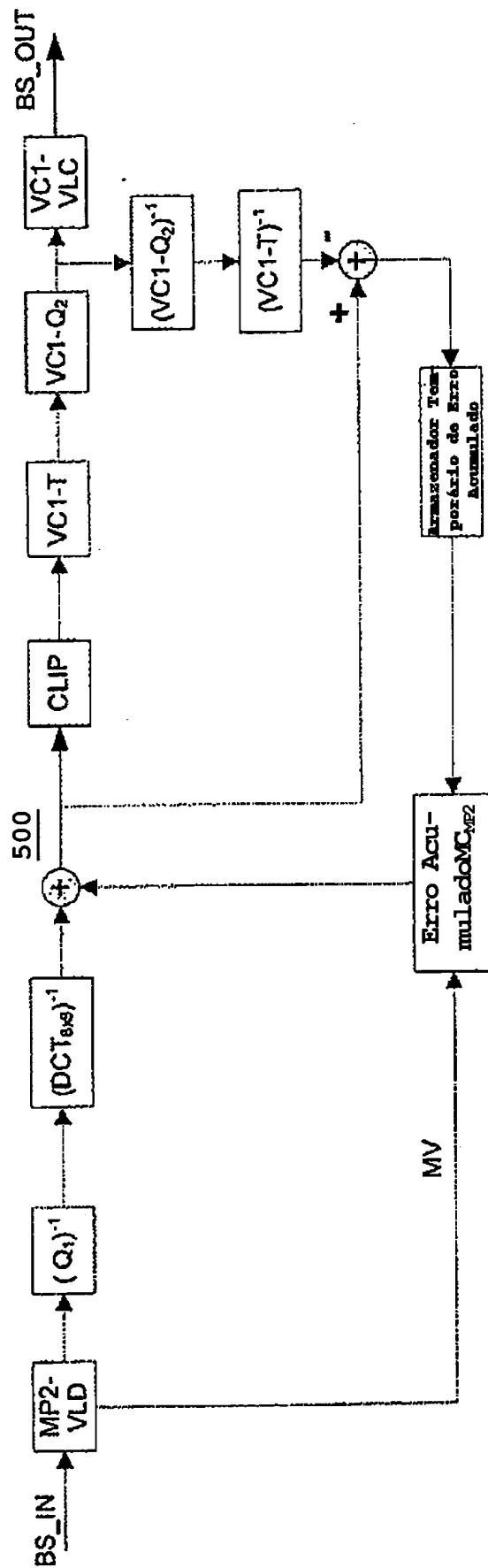


Fig. 5

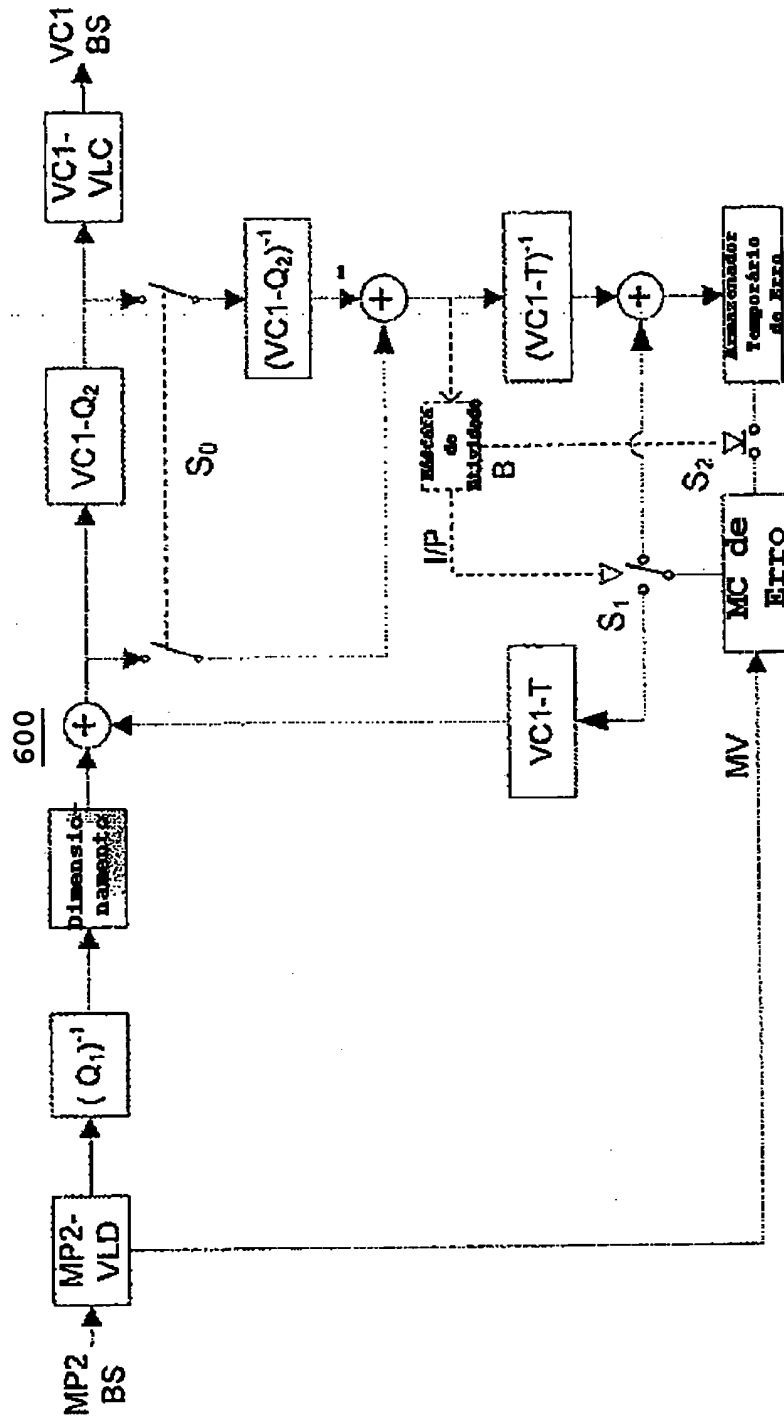


Fig. 6

700

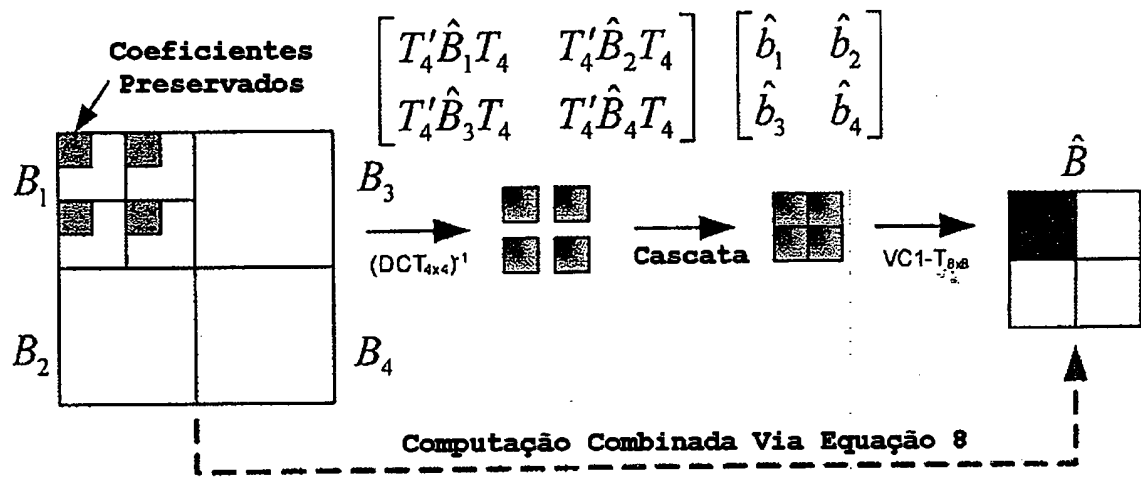


Fig. 7

800

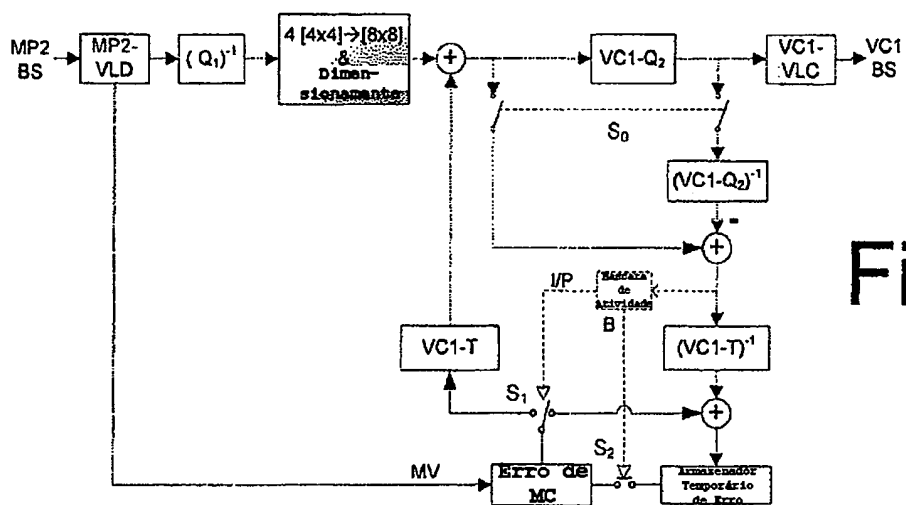


Fig. 8

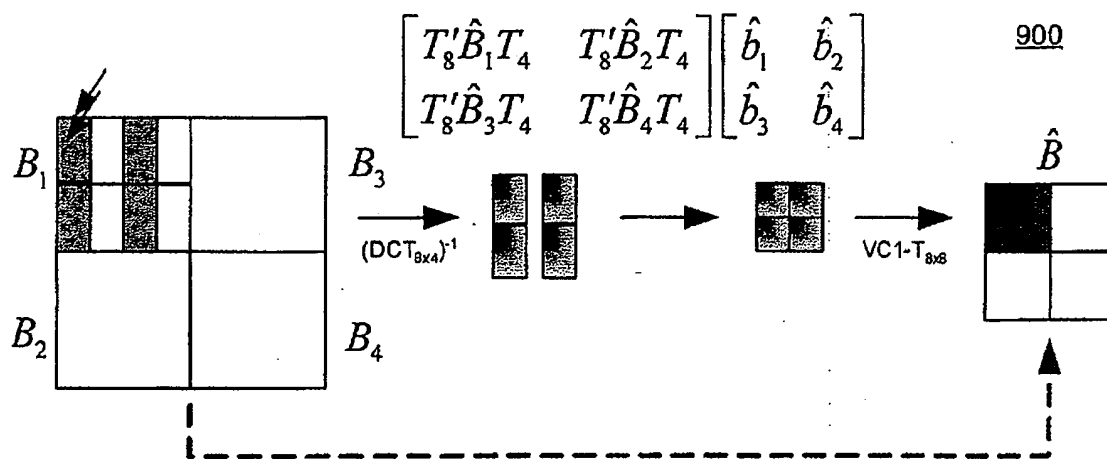


Fig. 9

1000

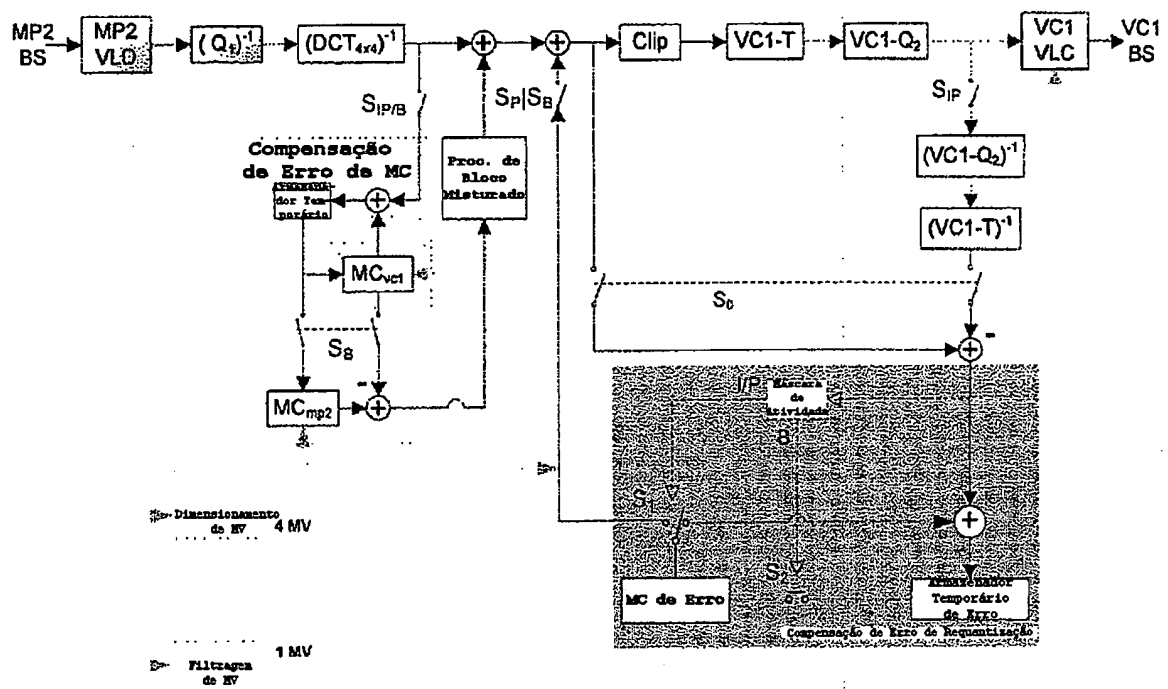


Fig. 10

1100

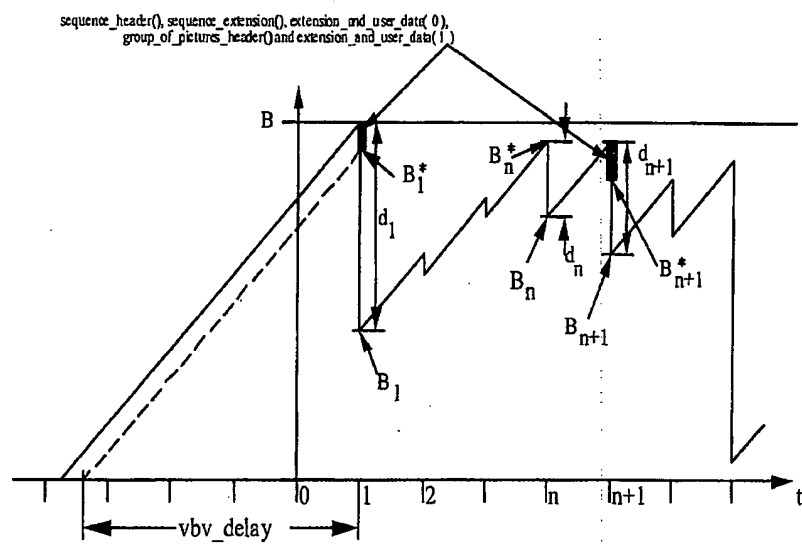


Fig. 11

1200

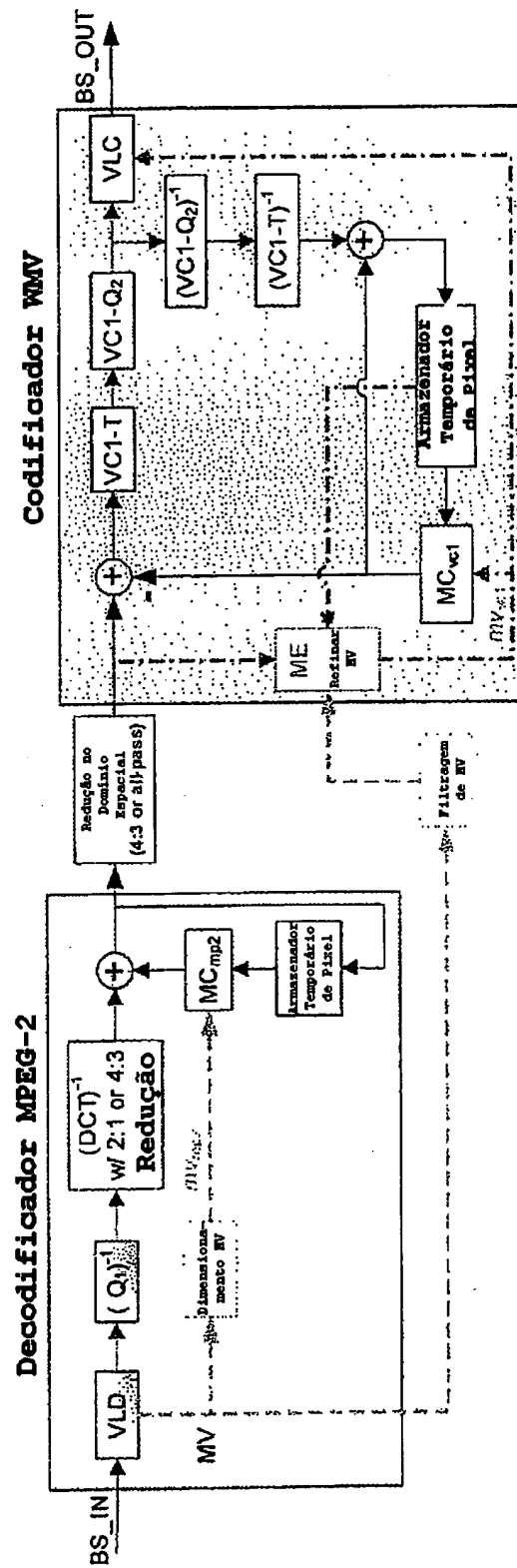


Fig. 12

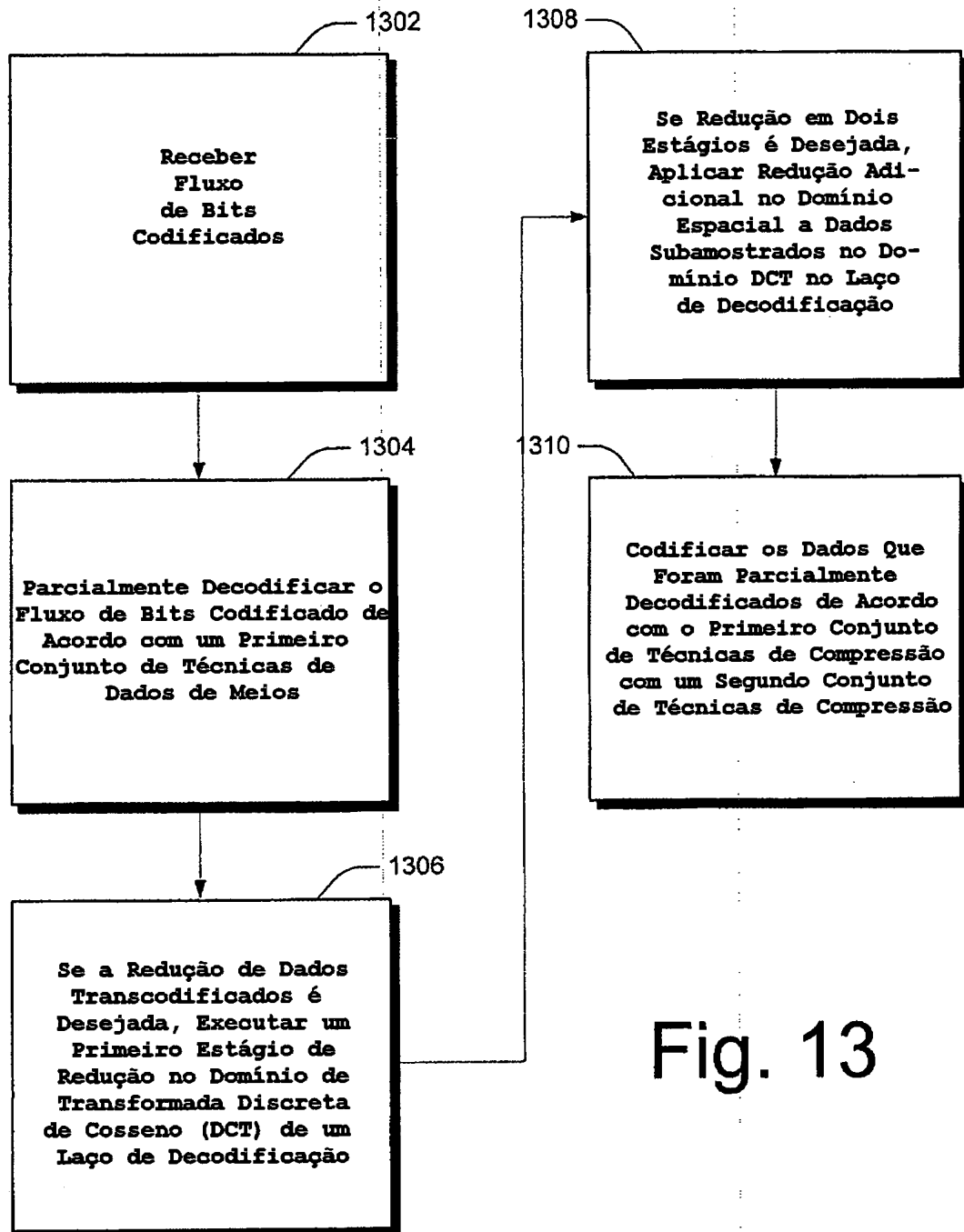
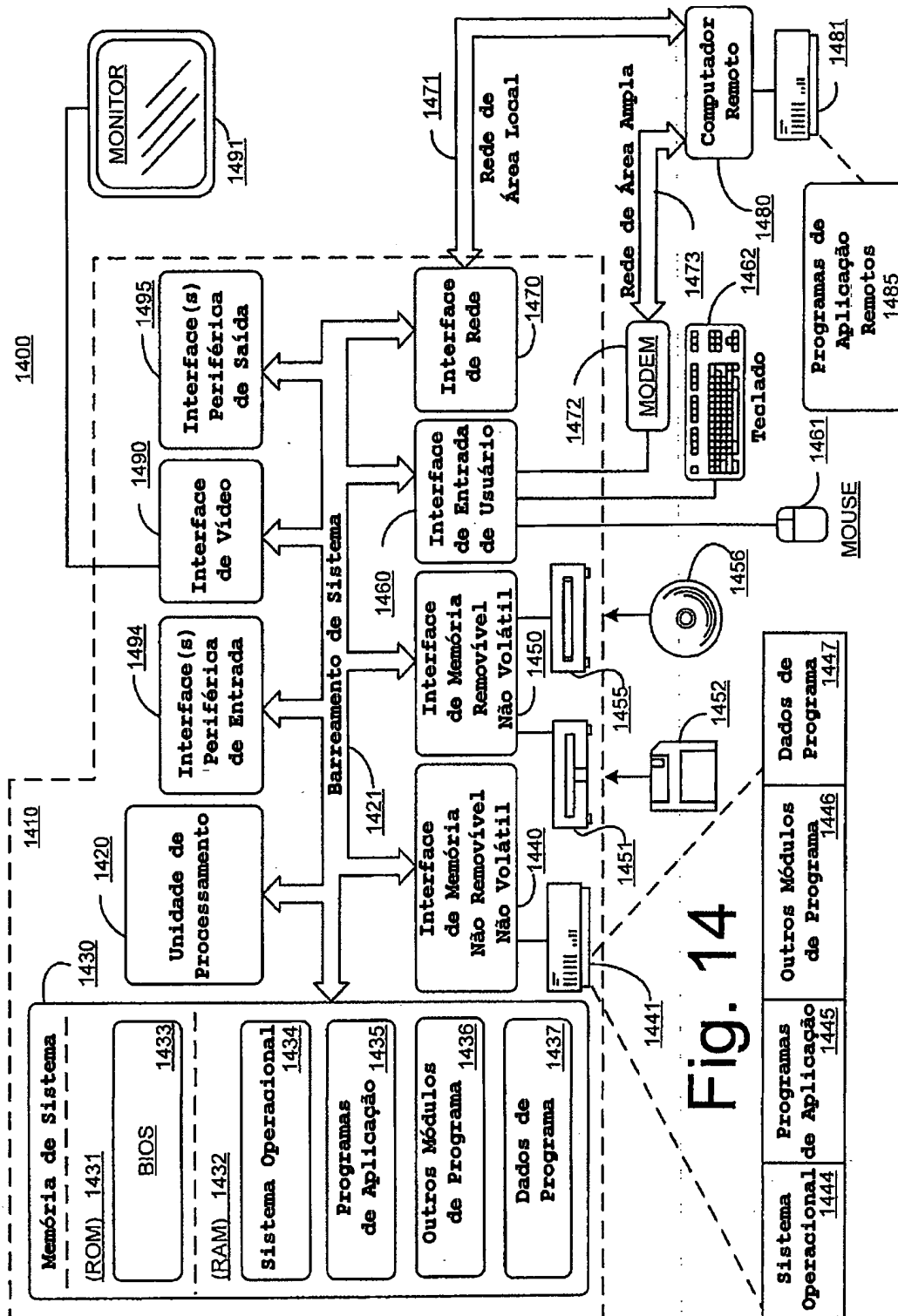


Fig. 13



Pt 0616034-4

RESUMO

"DECODIFICADOR DE SUBAMOSTRAGEM DE MUDANÇA DE RESOLUÇÃO ARBITRÁRIA"

5 A decodificação de subamostragem de mudança de resolução arbitrária é descrita. Em um aspecto, um fluxo de bits codificado é recebido. O fluxo de bits codificado é reduzido em um laço de decodificação no domínio DCT para gerar dados reduzidos.